ARI Research Note 97-01

VIEW: Visualization and Interactive Elicitation Workstation--A Tool for Representing the Commander's Mental Model of the Battlefield

Greg Zacharias, Christine Illgen,
Mehran Asdigha, and Jake Yara
Charles River Analytics

Eva Hudlicka
Psychometrix

Fort Leavenworth Research Unit
Stanley M. Halpin, Chief

December 1996

United States Army
Research Institute for the Behavioral and Social Sciences

Approved for public release; distribution is unlimited.

DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

U.S. ARMY RESEARCH INSTITUTE FOR THE BEHAVIORAL AND SOCIAL SCIENCES

A Field Operating Agency Under the Jurisdiction of the Deputy Chief of Staff for Personnel

EDGAR M. JOHNSON
Director

Research accomplished under contract
for the Department of the Army

Charles River Analytics

Technical review by

Jon J. Fallesen

NOTICES

DISTRIBUTION: This report has been cleared for release to the Defense Technical Information Center (DTIC) to comply with regulatory requirements. It has been given no primary distribution other than to DTIC and will be available only through DTIC or the National Technical Information Service (NTIS).

FINAL DISPOSITION: This report may be destroyed when it is no longer needed. Please do not return it to the U.S. Army Research Institute for the Behavioral and Social Sciences.

NOTE: The views, opinions, and findings in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other authorized documents.

REPORT DOCUMENTATION PAGE

1. REPORT DATE 1996, December	2. REPORT TYPE Final	3. DATES COVERED (from. . . to) March 1995-September 1995
4. TITLE AND SUBTITLE VIEW: Visualization and Interactive Elicitation Workstation—A Tool for Representing the Commander's Mental Model of the Battlefield		5a. CONTRACT OR GRANT NUMBER DASW01-95-C-0066 5b. PROGRAM ELEMENT NUMBER 0601102A
6. AUTHOR(S) Greg Zacharias, Christine Illgen, Mehran Asdigha, and Jake Yara (Charles River Analytics) and Eva Hudlicka (Psychometrix)		5c. PROJECT NUMBER B74F 5d. TASK NUMBER 6901 5e. WORK UNIT NUMBER C01
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Charles River Analytics, 55 Wheeler Street, Cambridge, MA 02138 Psychometrix, Arlington, VA		8. PERFORMING ORGANIZATION REPORT NUMBER R94371
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) U.S. Army Research Institute for the Behavioral and Social Sciences ATTN: PERI-RK 5001 Eisenhower Avenue Alexandria, VA 22333-5600		10. MONITOR ACRONYM ARI 11. MONITOR REPORT NUMBER Research Note 97-01
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.		
13. SUPPLEMENTARY NOTES COR: James W. Lussier		

14. ABSTRACT *(Maximum 200 words)*:

This research assessed development of a Visualization and Interactive Elicitation Workstation (VIEW) for inferring and representing mental models of the battlefield. The prototype is composed of two subsystems: a Visualization subsystem and a Knowledge Elicitation (KE) subsystem. The Visualization subsystem is composed of three interlinked modules: (1) a Tactical Visualization Interface providing liked displays for visualizing the battlefield; (2) an Object Database providing a common object representation for environmental and military entities; and (3) an Object World Model providing behaviors for objects in the database. The KE subsystem is composed of two linked modules: (1) the KE interface, which navigates across direct, indirect, and observational techniques, collection of elicited data, and KE analysis results; and (2) the KE Recording/Analysis Module, which records the elicited data. A demonstration illustrates coupling between the visualization and elicitation components and support of direct and indirect mental model examination. Direct techniques use a case-based interview format; indirect techniques include repertory grid analysis, multidimensional scaling, and hierarchical cluster analysis.

15. SUBJECT TERMS

Knowledge Elicitation   Visualization   Rapid visual prototyping

Mental model   Battlefield representation

SECURITY CLASSIFICATION OF			19. LIMITATION OF ABSTRACT	20. NUMBER OF PAGES	21. RESPONSIBLE PERSON (Name and Telephone Number)
16. REPORT Unclassified	17. ABSTRACT Unclassified	18. THIS PAGE Unclassified	Unlimited	99	

ACKNOWLEDGMENT

This work was performed under U.S. Army Contract DASW01-95-C-0066, as part of the Small Business Innovative Research (SBIR) Program. The authors wish to thank the Contracting Officer's Representative (COR), Dr. James W. Lussier of the U.S. Army Research Institute for the Behavioral and Social Sciences' Field Unit at Fort Leavenworth, KS, for his encouragement and continued technical and administrative support. We also thank our subject matter experts, Major Jody Daniels and Major Robert O'Brien, both from the U.S. Army Military Intelligence Corps, First Reserve Forces Intelligence School, Fort Devens, MA, for their support in scenario development and their contributions to knowledge elicitation sessions. Finally, we thank Ms. Kristy Shriver and Ms. Michele Maloney for their help in the creation and editing of this document.

VIEW: VISUALIZATION AND INTERACTIVE ELICITATION WORKSTATION—A TOOL FOR REPRESENTING THE COMMANDER'S MENTAL MODEL OF THE BATTLEFIELD

CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

Page

LIST OF FIGURES (Continued)

Page

LIST OF FIGURES (Continued)

Page

LIST OF FIGURES (Continued)

VIEW: VISUALIZATION AND INTERACTIVE ELICITATION WORKSTATION--A TOOL FOR REPRESENTING THE COMMANDER'S MENTAL MODEL OF THE BATTLEFIELD

Introduction

That mental models form the basis of complex human problem solving, decision making, and behavior has become an accepted credo in the cognitive psychology and cognitive science research community. While determining the exact nature of these representations remains an area of active research, evidence is accumulating that a variety of internal structures exist, representing different types of knowledge, at different levels of abstractions, in different formats, geared towards different tasks, and, in general, varying depending on the level of expertise (Chi, Glaser & Farr, 1988; Gentner & Stevens,1983; Johnson-Laird, Byrne & Schaeken, 1992). Research also indicates that a major factor that distinguishes experts from novices is the nature of their mental models and that expert decision-making and problem-solving thus depends on the rapid construction of flexible customized mental models that capture the critical features of the task at hand (Klein, Calderwood & Clinton-Cirocco, 1986; Larkin, 1983).

Planning and management of complex missions requiring the optimal use of multiple resources in real-time, such as that occurring during battlefield management, is a particularly critical area for mental model research. Because of the complexity of the problem-solving and decision-making in this domain, a wide variety of knowledge structures and inferencing processes must be used by the expert decision-maker to construct viable mission alternatives and to allocate resources in dynamic real-time situations. Knowledge of both enemy and friendly troop locations, resource distribution, area topography, strategic, tactical, and logistical constraints, as well as a large number of previous scenarios adapted to the current situation, are necessary to support effective decision-making and to enable complex *"what if"* mental simulations.

Since multiple experts with different, possibly conflicting, goals and perspectives must cooperate in devising a battle plan, it is essential that a common understanding of both the situation and the individual contributors' internal models be made explicit, so that all relevant factors can be taken into account in formulating the overall plan. Such understanding may be facilitated by visual displays of the task, context, and the participants' mental models. Currently, these displays are typically limited to a variety of maps and overlays, showing spatial static perspectives of the domain, and occasionally to PERT-style network representations indicating temporal dependencies among events and situations. While such visualizations are useful, they do not capture the complexity, dynamic nature, and richness of the human experts' mental model structures. An ability to rapidly display explicit visualization of the full-range of the participants' internal models would improve decision-making, facilitate shared understanding, foster the integration of multiple sources of expertise, and contribute to assuring fast and accurate situation assessment, resulting in a more efficient planning process and more effective plans.

While much progress has been made in mental model research, primarily due to the increasing ability to construct computational models of the inferred structures, thereby enabling better validation procedures, many challenges remain. Among the most critical ones are: 1) elicitation of knowledge which may not be directly accessible to conscious thought; 2) validation of the inferred models; and 3) design of display formats for mental model visualization.

The primary challenges in identifying mental models in battlefield command are therefore to:

- Devise techniques that can capture models not directly available to introspection
- Devise techniques that minimize distortions during the elicitation process

- Devise a set of methodologies that can assess the wide variety of representations used by different types of inferencing, at different skill levels, and for different tasks
- Design a set of displays for rapid visualization of the elicited mental models which can facilitate model refinement and support effective model sharing among decision makers
- Design an interactive, customizable user interface that can flexibly and rapidly display relevant models and show only information critical to the task at hand
- Develop a methodology for iterative model refinement and empirically-based validation

In light of these challenges, an open-loop approach to the problem - i.e., one where a model is inferred without subsequent empirical validation - has only limited utility, given the difficulties associated with model validation. We therefore have embarked on a hybrid approach to the problem of mental model identification, consisting of: 1) an empirical component focusing on the interactive elicitation of mental models from expert battlefield decision-makers; and 2) a computational modeling component focusing on the refinement and validation of the inferred models through further empirical testing of model-derived hypotheses.

Our Phase I effort has focused primarily on the design and prototyping of a Visualization and Interactive Elicitation Workstation (VIEW), to support iterative model refinement through the use of graphical model visualization tools and empirical knowledge elicitation techniques. Graphical displays representing the mental models can be designed based on initial data obtained from experts. These displays can then be embedded within an interactive graphical user interface (GUI) that facilitates the rapid display and manipulation of these structures, including visualization of a given situation from multiple perspectives, at different points in time, and at varying levels of granularity and abstraction. In other words, the visualization system attempts to capture the types of flexible manipulations of these structures that expert decision makers are able to perform with their mental models.

This Phase I effort was intended to demonstrate feasibility of the VIEW concept and set the stage for a Phase II effort which will focus on full-scope VIEW prototype development, empirical validation, and development of generic representational structures. Both Phases thus involve an iterative refinement approach to the problem of mental model identification: during Phase I this is accomplished through the use of graphical visualizations coupled with interactive knowledge elicitation techniques; during Phase II it is accomplished through hybrid empirical testing and successive refinement of representational structures.

Technical Objectives

The primary objective of the Phase I effort is to assess the feasibility of the VIEW concept design for mental model visualization, elicitation, and refinement. The objective is to demonstrate the use of the methodology for visualization and elicitation of the commander's mental model of the battlefield, at the brigade/battalion level. Basic questions to be addressed in the Phase I effort include:

- What is the structure and content of mental models of battlefield management and planning? How are different model types integrated (e.g., spatial and symbolic; static and dynamic)? To what extent is goal- and task-specific information integrated within these models?

- How do these models vary with task requirements? Are there differences in the types of perspectives of the situation, levels of abstraction, and degrees of granularity, stability, and general flexibility between task needs?
- What is the best visual representation format for each model? What are the best formats for displaying the processes (e.g., situation assessment, decisionmaking, planning, troubleshooting)? How should incomplete or uncertain information be displayed?
- Can we define an overall architecture for the VIEW prototype, that will be expandable from the Phase I demonstrator to the envisioned Phase II full-scope system? How should it be structured? What are the key modules?
- What is an appropriate software environment to ensure rapid prototyping capabilities, without incurring an expensive infrastructure of software development tools?
- What are the best parameters for manipulating the display format to enable rapid assimilation of information and to support decisionmaking activities? How should different display formats be integrated (e.g., static and dynamic)?
- Can we define a procedure-oriented methodology that will provide user guidance in the elicitation and visualization functions? How can we ensure that this will provide effective guidance in a Phase II effort?

By addressing these questions, we will be in a position to specify the requirements for a Phase II effort directed at full-scope development and validation of a VIEW prototype for mental model elicitation, visualization, and validation.

Technical Approach

The approach taken under this effort focuses on developing a concept design and demonstration prototype which integrates model elicitation and visualization for the battlefield commander. Six specific tasks compose our effort:

- Definition of Scope of Demonstration
- Review of Knowledge Elicitation Techniques and Software
- Review of Rapid Prototyping Visualization Software
- Development of VIEW Concept Prototype
- Demonstration of VIEW Concept Prototype
- Requirements Specification for Military/Commercial Development

We first defined the scope of the demonstration for this feasibility evaluation by reviewing military material such as Army Field Manuals and several documents from the Defense Technical Information Center. The subject matter of the material ranged from military intelligence and operations to mental models of commanders. By consulting our subject matter experts (SMEs), we developed a candidate scenario on which to focus our demonstration. Several scenarios were considered such as operations other than war and force-on-force offensive operations. We selected the force-on-force scenario because it provided an adequately constrained but sufficiently rich domain in which to demonstrate the functionality of the VIEW prototype. By conducting several follow-on knowledge elicitation sessions with our SMEs, we were then able to fine-tune our prototype to support knowledge elicitation functions.

We then reviewed knowledge elicitation techniques and tools and evaluated candidate techniques for implementation. A literature search was conducted specifically focusing on KE techniques that could directly support the specification and visualization of the commander's mental model of the battlefield. Both direct and indirect techniques were reviewed, and evaluated in terms of their ability to identify key components of the commander's model, their reliability, and their ease of use. In addition, we reviewed the availability and capability of associated software tools, to assess their potential for inclusion in a KE *toolkit*, to support computer-based elicitation sessions.

We then reviewed rapid prototyping visualization software options, for potential incorporation into the prototype. Based on a review of the visualization requirements called for in the demonstration, and a review of the KE requirements for commander mental model elicitation, we evaluated potential options for visualization software. The objective was to focus on packages which could be used for rapidly prototyping and displaying graphical objects, in an object-oriented environment that assures full connectivity between objects and their specific graphical visualizations.

We then developed a prototype visualization/elicitation tool, to support a demonstration of its use in the selected scenario. The prototype included specifications for interfaces to the KE tools selected for elicitation, as well as example visualization displays/controls implemented via the selected visualization software. An overall architecture was developed to integrate both the KE tools and the visualization software, and included a fully relational object-oriented data base to represent relevant objects and object sets composing the demonstration battlefield scenario.

We then demonstrated the prototype visualization/elicitation tool, to support an evaluation of system feasibility and potential utility in mental model formalization. Primary emphasis was in evaluation of the VIEW prototype's capabilities for visualizing different but related aspects of the tactical scenario at several different levels of organization and unit resolution. Effort was also devoted to evaluating the VIEW concept design in terms of its ability to support the interactive knowledge elicitation functions needed for mental model inferencing and representation. Functions not implemented in the Phase I VIEW prototype were identified and called out for follow-on Phase II development.

Finally, we specified requirements for military/commercial development of a full-scope tool and methodology for its use. For the military side, we focused on identifying further development and demonstration requirements to be met for a full-scope visualization/elicitation environment for commander mental model representation. For the commercial side, we identified promising commercial market areas, and particular market segments that could benefit from the development of a suitably specialized tool.

1.3 Summary of Results

The primary result of this study is a proof-of-concept demonstration of a visualization/elicitation prototype for graphically representing the mental models maintained by the battlefield commander.

The major study findings supporting this demonstration can be summarized in the following paragraphs:

A force-on-force offensive scenario was developed at three levels: brigade, battalion, and company. Our friendly brigade included assets such as mechanized armor and infantry while the

opposing brigade included mechanized armor. Included among the tools for scenario analysis were Decision Support Templates, Situation Templates, and Decision Trees. Courses of Action were examined for the friendly brigade and constituent battalions and led up to a high-intensity conflict with the enemy on a section of topography that involved river-crossings and the capture of a bridge.

We reviewed a variety of KE techniques, both direct and indirect. Both types of techniques are applicable to the mental model representation and visualization problem. However, no single technique or technique-type is adequate to capture the full scope of the internal representations. It is therefore necessary to use a repertoire of techniques in concert. In general, *case-based techniques* are preferred, because they quickly focus the discussion and generate concrete results (e.g., specific objects, specific decisions). Direct structured interviews are effective in eliciting a broad scope of knowledge but may not go deep enough to capture specific inferencing types or specific structures. The simple structured interview is thus best used in conjunction with a more specialized interviewing technique. Two techniques were found particularly well-suited for eliciting the commander's internal representations: a modification of Klein's *critical decision method* (1989b) which focuses on factors influencing a specific decision, and a *display-centered method* we developed during the course of this study, which focuses the interview process on both existing and desirable display formats.

The major disadvantage of the direct techniques is their limited capability to access knowledge which is not easily articulated by the expert in response to direct questioning. Indirect techniques do not require the expert to be able to directly access their knowledge, and thus represent an important complementary approach to elicitation which focuses on the more intuitive, idiosyncratic aspects of expertise. The two types of techniques are best used in conjunction: the direct techniques mapping out the broad scope of the knowledge structures and the indirect techniques allowing further focusing on specific constructs and substructures.

The review of visualization software for implementing the VIEW prototype focused on three operating systems: Unix/X-Windows, Macintosh OS, and DOS/Windows. Although exceptionally good graphics capabilities are supported by Unix machines, such as the Silicon Graphics Inc. Iris series, the relatively high price/performance ratios eliminated them from further consideration as potential hosts in what could eventually grow to be a large network of low-cost hosts. We thus favored the Macintosh OS and DOS/Windows environments. Although the former provides superior graphics tools, we selected the latter because of the much larger installed base at ARI, and the greater likelihood of integration/networking with existing Army systems.

With the focus on DOS/Windows-based software, we quickly identified four key software packages: Visual Basic for the interface, Microsoft Access for the database, Visio Technical for graphical support, and CLIPS for ruleset implementation. These applications all support Dynamic Data Exchange (DDE), so that the applications can be easily linked together. Since Windows is a multitasking system, many event-driven programs or applications are permitted to run concurrently. The DDE feature of Windows allows an application to directly and continuously exchange data with other Window-based applications that support DDE. Visual Basic is an object-oriented, Window-based programming language that facilitates the use of objects to initiate the execution of different programs and applications. Visual Basic uses the Microsoft Access database engine for its local data update and retrieval functionality. Visio Technical is a software package designed to run with Microsoft applications, and can be used to support development of the graphical interface. CLIPS is software developed at NASA's Johnson Space Center, and can

be used for implementing any formal rule set. It provides a rule/object-based environment in which to develop an expert system.

The VIEW system architecture is defined by two major subsystems: the Visualization Subsystem, and the Knowledge Elicitation (KE) Subsystem. The Visualization Subsystem is composed of three interlinked modules: the Tactical Visualization Interface, the Object Database, and the Object World Model. The Knowledge Elicitation Subsystem is composed of two modules: the KE Interface, and the KE Recording/Analysis Module.

The Tactical Visualization Interface supports the commander in two basic ways. First, it provides him with situation-relevant tactical information. Second, it provides him with the means of directly manipulating the object database, to create or modify the tactical situation. A graphical user interface supports navigation across a range of displays maintained in a display library.

The Object Database provides a common object representation for all visualization/elicitation components of the system, and is directly linked to the Tactical Visualization Interface via tactical commands generated by the user and object attributes sent to the displays. Three general classes of objects are maintained in the object class library: 1) terrain-related objects (terrain elevations, vegetation, roads, etc.; 2) military unit objects (echelons, types, weapons systems, etc.); and 3) ground environment objects (battlefield AO/AI, avenues of advance/approach, etc.).

The Object World Model supports an object-oriented simulation of both friendly and enemy forces operating over a specified battlefield reflecting weather and other environmental conditions. Linked to the Object Database via object commands and states, the module provides a direct means of dynamically modifying the database over time. An object behavior library supports the simulations of friendly/enemy mobility, and, via extension, wargaming capabilities.

The KE Interface supports the knowledge engineer in three ways. First, it provides a means of navigating among the KE techniques, via the control interface. Second, it supports the collection of elicited data from the commander who is interacting with the Visualization Subsystem. Finally, it provides on-line access to the results of KE analysis, to support interactive navigation among the displays, as a function of the results of the analysis. A graphical user interface supports navigation across a range of techniques maintained in a KE library.

The KE Recording/Analysis Module implements the actual recording and analysis of the elicited data via direct links to the KE Interface. In addition, to insure close linkage with the Visualization Subsystem, the recording modules also accepts as inputs the Visualization configurations selected by the commander via the Tactical Visualization Interface, as well as "snapshots" of the tactical situation as maintained by the Object World Model.

The VIEW prototype was implemented as a Visio Technical extension. The Visio extension approach to software development involved three interrelated steps. The first step involved creating a specific multi-window Visio workspace by modifying the Visio development environment to the specific requirements of this application. The term workspace here refers to a collection of interactive interfaces that are integrated based on a specific design and hierarchy. The second step consisted of adding functionality to the software and its host environment (i.e. the workspace) by embedding stand-alone and functionally independent executables in the environment itself. The stand-alone executables were developed in the Visual Basic development environment. This Windows-based package is very suitable for fast implementation of software designs that involve multiple interrelated interfaces. Furthermore, this development language has provisions for fast and easy access to databases created in the Microsoft Access application. In

addition to linking all objects in the workspace to the Microsoft Access databases, using Visual Basic for developing the executables also rendered the overall environment more flexible for the user. The third and final step in the development process involved adding functionality to various objects in the workspace by building stand-alone Visual Basic executables. These executables perform several types of tasks depending on the nature of the object they are linked to. For example, give the user access to different interfaces, as well as object attributes that would be otherwise hidden from the user. Through these stand-alone codes, object databases are updated whenever the user modifies an object attribute through any of the interactive interfaces.

Three aspects of the VIEW prototype are critical for its usefulness in mental model elicitation and visualization: the *variety of display formats* available to the commander, the ability to *navigate among these displays* in an unrestricted manner, and the ability to *query* the VIEW prototype and *highlight* display areas that satisfy particular parameters.

The VIEW prototype provides nine distinct display formats to capture the complexity of battlefield mental representations and mental models. The display formats include: maps and overlays, bar graphs, decision-trees, synchronization matrices, unit hierarchies, organization charts, and a variety of dialogue boxes and text windows. The basic display formats can be modified by the commander to reflect the specifics of a particular situation. Each display emphasizes a different combination of display/mental model parameters and thus different displays are suited for different types of inferencing and information integration. Examples of the individual display formats are described below.

A key display format in VIEW is the familiar *map and overlay* display, which is currently the predominant graphical format used externally by the army commanders. The combined map+overlay displays have a number of advantages: they represent a large amount of information in a readily understandable, familiar format; they combine spatial representations (which trigger lower-level perceptual processing) with abstract symbology (which trigger higher-level symbolic processing), thus providing both an overall context (e.g., map of an entire area) and a specific aspect of the situation on which to focus (e.g., arrows representing movement, icons representing units and weapons; etc.).

The *bar graph* represents an efficient and effective means of rapidly displaying the same type of information (e.g., remaining or required quantity) about a number of different variables (e.g., different resources). The format of the display lends itself to a fast assimilation of the relative status of a large number of variables and anomalies can be identified quickly and in a single scan.

While new display formats can capture a unique way of viewing information, in many cases an enhancement of an existing display format is sufficient to create a powerful means of filtering and combining relevant information. A *hierarchical depiction of the unit composition* is an example of such a display format. The familiar hierarchy provides an overall context, allowing the commander to view units at different levels of hierarchy in the same "scan", and providing a display background on which a variety of information (i.e., different characteristics of the particular unit) can be overlayed (e.g., weapons and resources available, level of combat readiness, etc.).

Another hierarchical display, the *decision tree*, is unique in that it combines a trace of a cognitive process over time; namely, it provides a trace of the decision making process with respect to the development of a particular COA sequence. Time is thus an implicit dimension in this display. Furthermore, the display is highly abstract and symbolic, depicting a series of

complex situations by a single labeled node in a tree diagram. As such, this display is well suited as a type of navigation backbone, through which to access the variety of other displays and information available about the situation.

The *navigation component* of the prototype facilitates unrestricted movement between the different display formats by allowing the commander to view displays containing identical objects or displays depicting related relevant information.

A critical component of VIEW prototype is the support it provides for *automatic detection of specific conditions* of the terrain, units, resources, or overall situation that might be of interest during planning. These conditions are expressed either as queries to the system or as rules defining some alarm or alert condition or a general situation of interest. Queries and rules are used to represent situations that might be desirable or undesirable and are a means of automatically detecting particular situations and displaying relevant information to the commander. Queries and rules thus serve the function of an *intelligent assistant,* who is aware of particular conditions which the commander should be aware of and notifies the commander when conditions occur. In the VIEW prototype the queries and rules thus allow the commander to explicitly visually represent important tactical decision making information combined into a single high-level construct. Examples of such constructs were elicited from the SMEs using repertory grid analysis.

The design of the VIEW prototype provides the knowledge engineer with a wide variety of tools to support the process of knowledge elicitation, the subsequent data analysis, and the final interpretation of the results, where necessary. The VIEW design provides an environment within which a variety of knowledge elicitation techniques can be performed, both direct and indirect, and a variety of data collection methods can be employed to support these techniques. The knowledge elicitation component of the design is tightly coupled with the visualization component, and thus the full-functionality of the visualization component is available to the knowledge engineer and the subject matter expert. The user (knowledge engineer or subject matter expert) interacts with the VIEW prototype via graphical user interface, which contains a number of screens that support a variety of knowledge elicitation techniques. The existing design demonstrates a sequence of mock-up interface screens and indicates how these would be used during an elicitation session.

Specifically, the VIEW elicitation design provides the user with a variety of graphical user interfaces. The prototype design includes the following functionalities:

Graphical Displays and Visualizations

• A library of graphical displays at varying levels of complexity which can support both direct and indirect elicitation.

• Support for a variety of data collection techniques through the systematic presentation of displays and stimuli to the SME to elicit both qualitative and quantitative judgments.

Direct Elicitation Techniques

• Facilities for entering and analyzing free-form text while viewing different displays for a particular scenario.

• Facilities for constructing and editing domain vocabularies and concept maps during the elicitation session.

• Facilities for constructing aggregate structures from these domain primitives to reflect the experts' mental models.

• Facilities for editing and browsing the elicited structures.

Indirect Elicitation Techniques

• Facilities for editing and transformation of the elicited data.

• A repertoire of statistical techniques for analysis.

• A flexible environment for displaying the analyzed data and for assisting with the interpretation process.

The direct knowledge elicitation techniques, case-based display-centered interviews and decision-centered interviews, all provided the data for defining the critical elements of the visualization architecture: object definitions (e.g., terrain, terrain types, environmental objects, map overlays, military templates for depicting situations and decision-making, etc.), display definitions (e.g., maps and overlays, synchronization matrices, decision trees, bar graphs, process diagrams, etc.), and query and rule definitions (e.g., definitions of specific constraints representing high-level cognitive and perceptual constructs of interest to the commander). In addition to these data, the display-centered techniques provided information about the desired types of displays and their use during battlefield visualization. Examples of desired display types and functionalities included the following: ability to view a 3-D terrain representation from arbitrary perspectives, ability to combine and display a variety of weapons and electronic equipment characteristics, ability to support wargaming and what-if simulations through animation, automatic overlay and comparison of event and situation templates to quickly detect differences between predicted and actual situations, and the ability to zoom within an area and rapidly move among different levels of abstraction. Due to the limited scope of this initial effort many of the suggested display formats and display manipulations could not be implemented. However, the information generated using the display-centered elicitation method is included in the recommendations for the follow-on Phase II effort.

The indirect knowledge elicitation effort, which focused on repertory grid analysis, yielded a number of classification attributes relevant to battlefield visualization. These attributes were elicited using different courses of action and different corridors of mobility. Examples of elicited classification attributes are: fire power deployable, concealment sensitivity to season, possibility of destroying concealment, maneuverability in bad weather, maneuverability in reduced visibility, safety in reduced visibility, vulnerability to ambushes, areas of vulnerability within corridor, ability to conceal rate of movement, ability to conceal number of troops, and ability to conceal exact location.

While some of the attributes were also obtained through direct elicitation, the repertory grid method generated a large number of complex constructs quickly and easily. We therefore recommend it as an effective and efficient means of obtaining complex cognitive and perceptual constructs. Our experience with using just two entity types for comparison and generating over 60 attributes, many of which represent complex tactical constructs, indicates that repertory grid analysis is a powerful technique for eliciting the commander's mental model attributes and warrants further exploration. A major feature of the elicitation component of the VIEW prototype design is a flexible means of presenting graphical entities for comparison during the initial stages of the repertory grid process. The VIEW prototype thus promises to be a powerful tool for eliciting a wide variety of tactical constructs, which can then be translated into visual format using the visualization component of the VIEW prototype.

Following our prototype demonstration, we specified the requirements for full-scope development of the VIEW concept, under a Phase II design, development, and validation effort. Under Phase I, the objective was to establish feasibility; under Phase II we would considerably expand the scope, increase the functionality of the modules, and fully explore the tool's utility in a formal validation exercise. The system architecture would follow that established by this Phase I study, but the functionality of the individual component modules would be considerably expanded. In particular, the *object world model* would be expanded to provide for dynamic simulation of friendly/enemy mobility, and limited computer-based wargaming. The object database would undergo considerable expansion in both the types of objects represented, and in the fidelity of representation. This would include all three object classes now represented in the Phase I model: terrain objects, military unit objects, and ground environment (operational) objects. The *visualization module* would also be expanded, to account for a greater range of conventional military displays, as well as an expandable set of unconventional displays subserving effective mental model representation. The *knowledge elicitation* module would be extended considerably beyond the user interface design, and include full functionality both in the interface, and in the underlying analysis software libraries. A direct linkage to the object database would also ensure that a "snapshot" of the actual tactical situation was available, to support the development of context-dependent user activity models.

We believe that these results demonstrate the basic features of the VIEW concept for mental model visualization, elicitation, and refinement, particularly as applied to the commander's mental model of the battlefield. The study was specifically structured to be narrow in scope, but of sufficient depth to ensure the reliable specification of requirements for a full-scope system.

1.4 Report Outline

Chapter 2 provides technical background on past research and current technologies most relevant to our effort to develop a VIEW concept prototype. Section 2.1 provides a brief overview of key mental model research, while section 2.2 reviews relevant work in tactical situation assessment and decisionmaking. Section 2.3 reviews relevant direct and indirect knowledge elicitation techniques, and identifies shortcomings in some techniques that might be proposed for this domain. Finally, section 2.4 identifies visualization software which can be used effectively for the prototype development effort.

Chapter 3 provides a functional description of the VIEW prototype. Section 3.1 defines the system architecture and provides a general overview of system functionality. Sections 3.2 and 3.3 then describe the two key subsystems, the visualization subsystem and the knowledge elicitation subsystem, respectively.

Chapter 4 describes operations of the VIEW design, and illustrates capabilities of the prototype system. Section 4.1 provides an overview of the visualization/elicitation process to place the VIEW functions in context. Section 4.2 then describes a sample tactical scenario used to focus the KE effort and the prototype development effort. Section 4.3 proceeds with an example visualization session conducted by the commander during mission planning, to illustrate VIEW visualization functionality. Section 4.4 complements this with a description of some of the knowledge elicitation sessions conducted to support the VIEW prototype development effort.

Chapter 5 summarizes the key tasks conducted under this effort, presents the major conclusions, and outlines the recommendations for a Phase II development effort.

2. Background

This chapter provides technical background on past research and current technologies most relevant to our effort to develop a VIEW concept prototype. Section 2.1 provides a brief overview of key mental model research, while section 2.2 reviews relevant work in tactical situation assessment and decisionmaking. Section 2.3 reviews relevant direct and indirect knowledge elicitation techniques, and identifies shortcomings in some techniques that might be proposed for this domain. Finally, section 2.4 identifies visualization software which can be used effectively for the prototype development effort.

2.1 Mental Model Research

Given the fact that there is no consistent definition of what exactly a *mental model* is, and that the precise meaning of this term varies depending on the situation, we propose the following working definition of the term for the contemplated effort: A mental model is a task and *situation-specific* representation that supports problem-solving and decision-making in a particular context. A variety of such representations exists for any given problem-solving situation, supporting different types of inferencing, depending on the task at hand. These representations are dynamically constructed from a related set of underlying knowledge structures which contain both more general abstract knowledge, and a repository of highly-specific cases. Information relevant for the task at hand is dynamically extracted from these underlying structures during problem-solving. Figure 1 illustrates the relationship among these structures, in the context of different types of memory and processing. Figure 2 illustrates examples of specific visualizations a battlefield commander might employ to support tactical decisionmaking.

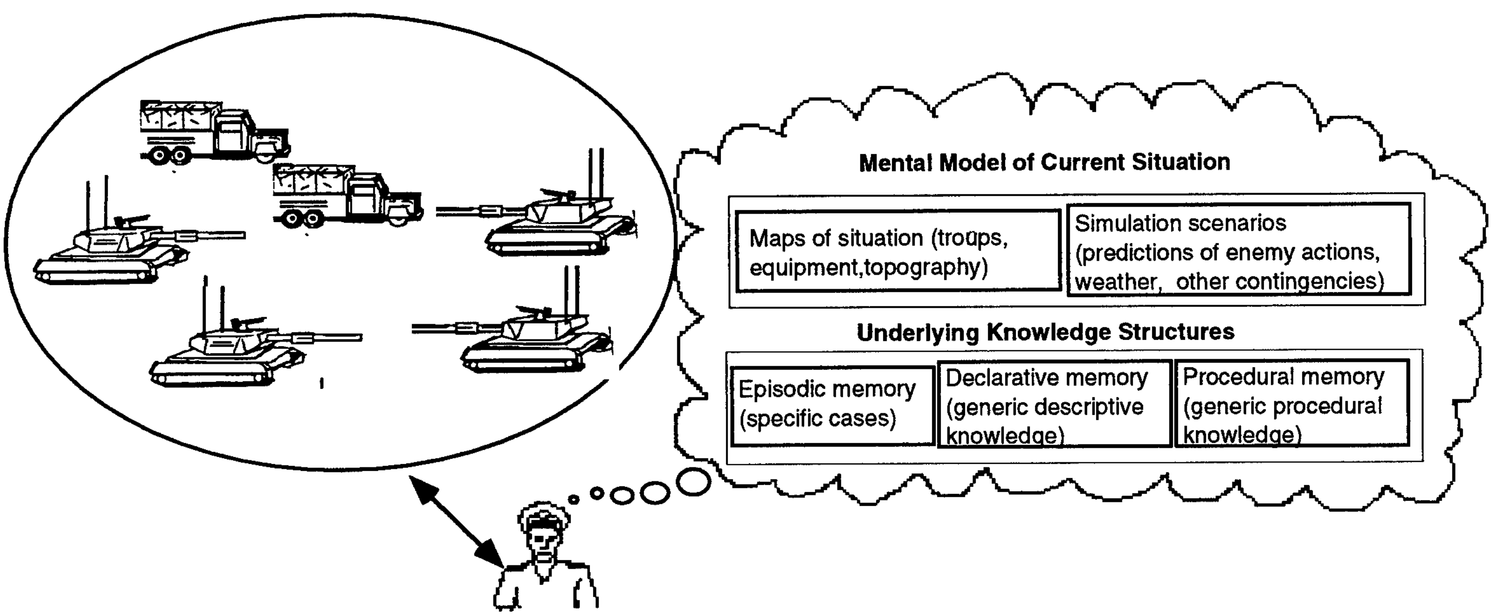


Figure 1. Mental model of tactical situation

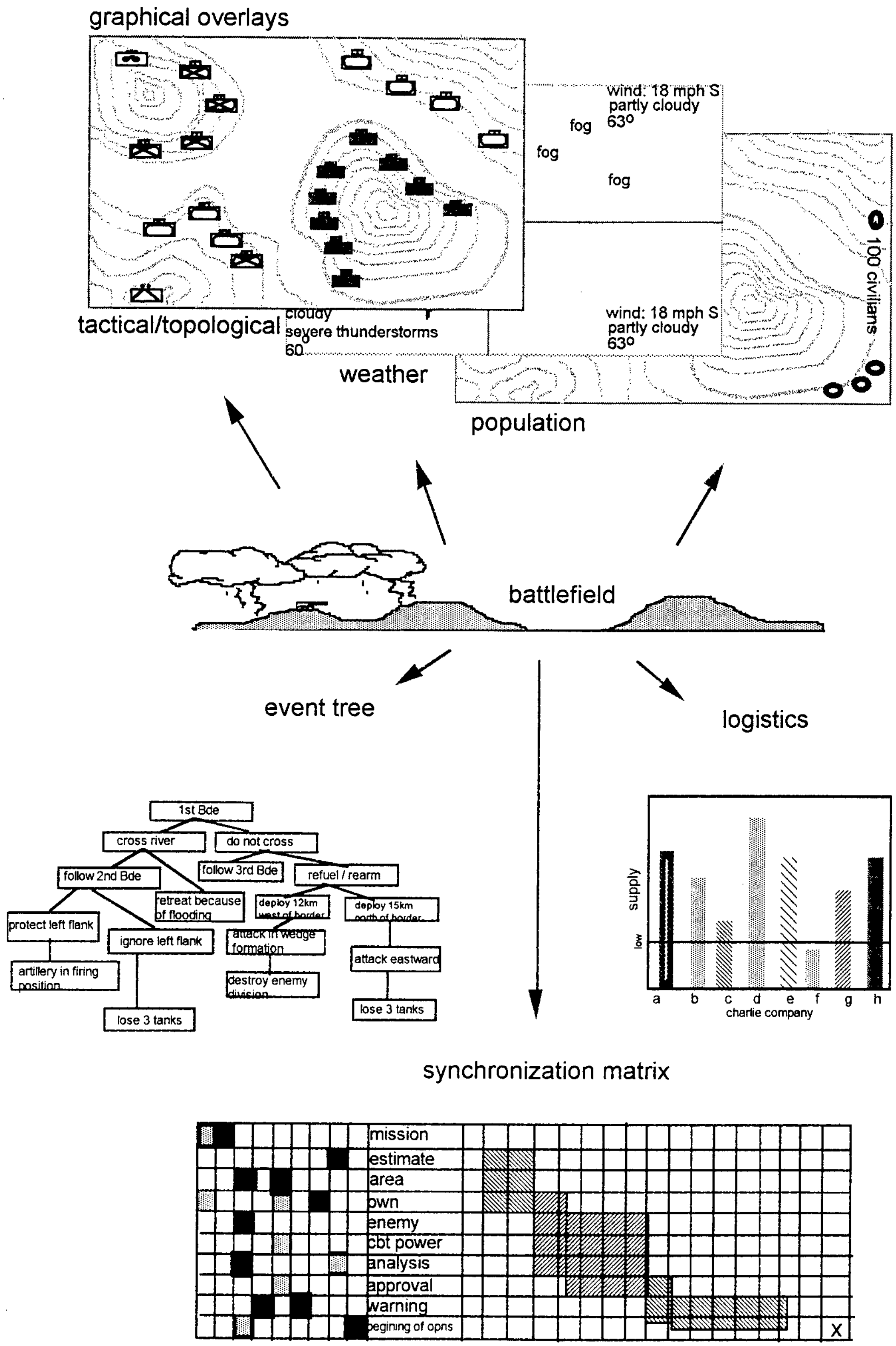


Figure 2. Multiple-format representation of a battlefield situation.

Due to the variety of mental models required to support battlefield management decisions, it is clear that different elicitation techniques are necessary to capture the variety of internal representational structures, and that these must be applied in a variety of contexts and tasks. While there are many open issues regarding the exact nature of mental models and their relationship to the underlying knowledge structures, recent research in cognitive science and psychology indicates that similar representational structures are involved in both cases. The consequence of this finding is that a similar set of techniques is therefore appropriate to identify the nature of these structures.

The study of mental models is relatively new, enabled by a confluence of cognitive psychology and AI, and, more recently, cognitive science, which provide unique tools for combined computational modeling and empirical studies. A variety of motivations, domains, theoretical assumptions, and, consequently, methods exist for mental model research and the area is not yet mature, as evidenced by the fact that the term *mental model* itself is not used consistently through the literature and encompasses a range of internal structures. In the remainder of this section we summarize state-of-the-art of mental model research and highlight important results. We focus here on mental models of complex cognitive/perceptual tasks, relevant to the proposed effort, rather than earlier simpler studies of sensorimotor activities.

Motivations for mental model research include both theoretical interest in human information processing and deep domain theories of knowledge (Gentner & Stevens, 1983; Johnson-Laird et al., 1992), construction of knowledge-based expert systems (Berry, 1987; Boose, 1985; Cooke & McDonald, 1988), decision making research (Klein et al., 1986), and applications of these findings for training, understanding human errors, communication, and design, in the context of human-centered automation. Depending on the specific goals, different domains and tasks have been studied.

The most widely studied domains have been relatively simple mechanical and electrical devices (e.g., calculators, simple electronic circuits), simple physical systems (e.g., bodies in motion, behavior of liquids) (Roschelle & Greeno, 1987), or concepts (e.g., electricity) (Forbus, 1983; White & Frederiksen, 1987; Gentner & Gentnet, 1983; deKleer & Brown, 1983). These relatively simple domains provide contexts where both the domain and the problem-solving are well-understood, and, as such, provide good context within which to study how humans construct internal representations. Since the advent of knowledge-based systems, more complex domains have begun to be addressed in mental model research in order to elicit the expertise of human experts. In this context a wide variety of domains have been explored, including computer programming, medical diagnosis, complex system troubleshooting, image understanding and interpretation, and decision making in a number of domains including law, aircraft inspection, fire fighting, military tactical decision making, and battlefield management (Chi et al., 1988; Hudlicka & Huggins, 1994; Klein et al., 1986; Broadbent, Fitzgerald & Broadbent, 1986).

The typical methodologies include some combination of knowledge elicitation techniques to obtain data from human problem solvers, and a computational modeling approach wherein the elicited model is implemented using artificial intelligence techniques to determine whether it can account for observed empirical data (Forbus, 1983; Kieras, 1984; Detterman, 1989; Hegerty, Just & Morrison, 1988). The elicitation techniques used most often have been simple observation studies, questionnaires and protocol analyses (Ericsson & Simon, 1984), and a variety of specialized techniques such as critical decision method (Klein et al., 1986), from which the experimenter reconstructs the model. Recently, indirect techniques such as repertory grid analysis,

multi-dimensional scaling, and hierarchical clustering have been used in this research area (Hudlicka & Huggins, 1994; Olson & Biolsi, 1990).

The primary result of these studies is the following set of observations: 1) mental models take a wide variety of forms, including spatial (Gentner & Stevens, 1983), propositional (Johnson-Laird et al., 1992), and combined network representations (Roschelle & Greeno, 1987; deKleer & Brown, 1983), depending on the task and the subject's level of expertise (Larkin, 1983; Chi et al., 1988), 2) individual mental models interact during problem solving (Roschelle & Greeno, 1987), and 3) that mental models do not always represent distinct, completely accurate and unambiguous structures (Norman, 1983). A critical finding is the apparently ubiquitous use of *qualitative reasoning* in many tasks involving the reasoning about dynamical systems (Hegerty et al., 1988; Roschelle & Greeno, 1987; deKleer & Brown, 1983). Qualitative reasoning represents an abstraction of the system behavior that allows the experts to quickly perform simulations (envisionments) of different system states under different conditions.

A number of controversies regarding mental model representations and reasoning exist. These include the *imagery debate* regarding the nature of mental representations and the role of perceptual processes in manipulating and interpreting these images: to what extent do humans manipulate and interpret actual perceptual analogs of real images and to what extent are abstract perceptual images constructed to support problem solving (Kosslyn & Schwartz, 1977; Kosslyn, Cave, Forbes & Brunn, 1983). Another debate concerns the basic nature of decision-making and problem-solving: are these processes the result of a complex search through an explicit problem space supported by generative mental models (deKleer & Brown, 1983; Forbus, 1983), or are they the result of complex one-shot recognitions and interpretations of a situation (Klein, 1989a)?

2.2 Tactical Situation Awareness and Decision Making Behavior

Human performance in decision-making in general, and in tactical planning in particular, have been studied extensively by psychologists and human factors researchers, primarily through empirical studies in the field but increasingly so with computational modeling tools. These studies span the theoretical-to-applied spectrum and cover many domains. Many aspects of human performance have been studied. Endsley (1995) and Adams, Tenney & Pew (1995) discuss a psychological model of decision-making, focusing in particular on situation awareness (SA), and the impact of particular system characteristics on the operator workload, attention and memory requirements, and the likelihood of errors. Klein (1989b, 1987) has studied a particular type of decision-making predicated on the quick extraction of salient cues from a complex environment and a mapping of these cues to a set of procedures. Research indicates that such Recognition-Primed Decisionmaking (RPD) plays a major role in tactical planning and it is therefore critical for decision-aiding systems to recognize this mode of human information processing and to support it through appropriate display design (Brezovic, Klein & Thorsden, 1987). Studies have been conducted investigating reasoning styles and comparing analytical and intuitive cognitive styles in expert decision making (Hammond, Hamm, Grassia & Pearson, 1987). Results indicate that particular attributes of tasks (e.g., number of redundant cues, complexity of the situation, and degree of perceptual vs. abstract and objective task elements) induce an automatic method of tabulating underlying judgments. Such results are particularly relevant for tactical visualization, where complex combinations of intuitive and analytical judgments and decision-making are common in assessing the situation.

In the battlefield management and tactical planning domain, a number of studies of human performance have been conducted. Several findings stand out in their relevance to tactical situation visualization and analysis. Several types of biases in tactical SA have been identified (Tolcott, Marvin & Lehner, 1989; Fallesen, 1993) which contribute to the inadequate development of tactical alternatives or to the selection of an inappropriate final COA. Of particular importance is the *primacy bias*, that is, selecting an *a priori* option and then looking for confirmatory evidence and ignoring disconfirming evidence for that option. Another common bias is *success orientation*, that is, the overconfidence in friendly plans and underestimation of possible enemy activities that could jeopardize projected friendly activities (Fallesen & Michel, 1991; Lussier, Solick & Keene, 1992).

Empirical studies of tactical planning and decisionmaking indicate that certain categories of failures are common, resulting in inadequate COA development and selection (Fallesen, 1993). Fallesen (1993) divides these failures into categories according to the stage of the COA development process (e.g., situation assessment, formulation of alternative COAs, comparison of these alternatives, wargaming, etc.). For each category he then identifies the most critical factors that contribute to ineffective and non-optimal performance. Examples of these factors are failures to use systematic comparison strategies for alternative COAs, failures to verify uncertain information, failures to develop adequate action/reaction trees due to inadequate wargaming, failures to consider all factors, failures to verify assumptions, failures to assess information quality, failures to interpret available information, and failures to make predictions for situation assessment. Other research indicates that knowledge of enemy activities is particularly critical and often neglected by tactical planners (Shaw & Powerll, 1989; Castro, Hicks, Ervin & Halpin, 1992).

A number of studies have been conducted focusing on the differences between expert and non-expert performance. An experiment designed to determine differences in information usage by tactical planners indicated that 78% of critical facts identified by the experts were missed by the non-experts. The facts missed by non-experts included timing information, actions of adjacent units, changes in boundaries, enemy activities, terrain constraints, mobility, engineering capabilities, and logistical loads (Fallesen et al., 1992). Another critical difference between experts and non-experts is the use of uncertain information. Experts were more aware of uncertain assumptions and made explicit predictions of events that would confirm their expectations and thus confirm or disconfirm assumptions (Tolcott et al., 1989). A study of expert military tactical decision-making (Deckert, Entin, Entin, MacMillan & Serfaty, 1994) found that experts' performance differed along a number of dimensions, including awareness of enemy activities, learning from past mistakes, flexibility of planning, seeking of disconfirming evidence, deeper exploration of options, and better management of uncertain information.

2.3 Knowledge Elicitation and Psychometric Techniques

The knowledge elicitation (KE) component of this study depends on the use of a set of psychometric techniques, each designed to access a particular type of internal representation. Since the ability to directly verbalize models varies greatly and because many studies report difficulties associated with attempts to verbalize mental models, particularly in the case of experts, we propose to use indirect knowledge elicitation techniques to augment more conventional direct (or introspective) techniques as a means of accessing these structures. Indirect techniques, adapted from experimental psychology and memory research, are designed to overcome the

limitations of purely direct techniques in accessing all relevant knowledge and processes and to minimize the possibility of distortion of the data that exists with direct techniques (Ericsson & Simon, 1984; Knaueper & Rouse, 1984). The focus of the proposed techniques is the elicitation of perceptual attributes characterizing important entities in the domain of interest. As such, these techniques are particularly relevant for knowledge elicitation in situations characterized by recognition-primed decision-making (Klein et al., 1986), which is thought to be mediated by complex perceptual-cognitive features.

Knowledge elicitation (KE) techniques have been developed by psychologists and artificial intelligence researchers to access human knowledge structures, whether in the context of memory and expertise research (Olson & Biolsi, 1990), or in the more applied setting of knowledge-based system construction (Gaines & Boose, 1988) and mental model research (Klein, 1989a; Rouse & Miller, 1986).

Many techniques exist, but they can be roughly split into two categories: direct and indirect. Direct techniques, such as interviews and protocol analysis, are based on the assumption that the experts or subjects are able to directly articulate the knowledge they use in problem-solving and decision-making. While the direct techniques are powerful methodologies for knowledge elicitation and knowledge acquisition they have several drawbacks as techniques for obtaining the expert's underlying mental models and details of the reasoning processes. *First,* only data accessible to conscious awareness can be reported. There is an on-going debate as to whether the data reported in fact represent the *actual* underlying thought processes or whether they are reconstructed by the expert and have little to do with the actual mental models and processes[1] . Psychological literature contains many experiments reporting exactly such reconstructions, the best known one being the work of Nisbett & Wilson (1977). There is evidence that truly expert knowledge is difficult to articulate and that what is being reported by the expert is at best *intermediate* level of reasoning (Schmidt, Boshuizen & Hobus, 1988; Berry, 1987).

Second, even if we accept introspection as a reliable means of accessing internal processing, direct verbal techniques have applicability only in situations where expert problem-solving is verbally mediated or at least when it can be expressed in terms of language. This is not typically the case for tasks which rely on perceptual and motor processing, which are often performed on an almost reflexive basis, and are difficult, if not impossible, to articulate. To the extent that such processing forms the basis of higher-level reasoning, such as that used in complex tactical pattern recognition in battlefield management, a different set of techniques must be used to augment the purely verbal ones.

Indirect techniques represent an alternative set of methodologies for knowledge elicitation, which do not rely on the assumption that expert knowledge is directly accessible to conscious thought. Rather they assume that relevant knowledge is often not easily or directly accessible to conscious thought and cannot therefore be revealed by simple introspection in response to direct questions. The indirect methods attempt to by-pass this limitation by accessing pieces of the internal structures through a series of simpler questions, for example, through similarity judgments among items of interest, and from these data then reconstruct mental model structures and infer

---

[1] Note that when we question whether data are accessible via introspection we are speaking here about the detailed mental models and reasoning, not about the basic, general knowledge of the task and the domain, which can clearly be articulated and obtained via interviews.

the underlying knowledge and reasoning. These techniques do not rely on introspection, nor are they limited to data that are easily verbalized, as we shall see below.

2.3.1 Direct Knowledge Elicitation Techniques. All techniques that fall within this category rely on the experts' ability to directly articulate their knowledge of the subject matter and describe their decision-making and problem-solving processes. The techniques vary in the degree of structure they impose on the expert during the interviews, the degree to which the questions are open-ended, the types of questions asked, the means of recording the experts' responses, and the environment in which the sessions are conducted. Direct techniques fall into two broad categories: interviews and protocol analysis.

2.3.1.1 Structured and Unstructured Interviews. Interview is the simplest means of obtaining the experts' knowledge. During an interview the experts are asked a series of questions about the domain and the tasks. The interview is typically recorded and transcribed for later analysis. The questions range from the free form "Describe a typical battlefield management task" to the quite specific "What are the factors that contributed to your decision to attack the bridge from the south?" The most important characteristic of interviews is that they are retrospective; that is, the expert is asked questions about the subject matter, tasks, or decisions made in the past. Interviews are thus conducted "off-line" and not while the expert is performing his or her task. This can be an advantage, since the distance from the actual environment may allow some features of the problem-solving process to emerge. It may also be a drawback, since the expert may not be able to readily access all the reasoning and knowledge that takes place during an actual performance of a task.

A number of variations exist within the general category of interviews; the questions can be geared toward eliciting particular type and form of knowledge (e.g., causal models, taxonomies of entities in the domain, goal and procedure trees, etc.); the discussion may be focused on a specific case or scenario or it may be more general, spanning the domain as a whole; information about particular subtasks may be elicited by focusing the questions on such tasks (e.g., specific decisions made). Some representative specialized interviewing techniques are described below.

Inferential flow analysis is a specialized method of interviewing designed to obtain inferential or causal models of the experts' reasoning (Salter, 1983). This technique involves asking the expert a series of "why," "how," "what causes this to happen," "what if," and "what typically follows this" questions, in order to construct a diagram representing the expert's chain of reasoning about a particular problem. The results of inferential analysis are dependency diagrams among domain entities, which capture the structure of important domain or problem-solving processes. The entities comprising these models vary, as do the relations that link them. Thus a variety of causal models can be elicited. For example, if the entities are different situations on the battlefield, then the elicited network will represent the space of possible evolving situations over time and can be used as the basis of wargaming or what-if simulations. If the entities represent different steps in the battlefield management process, then the elicited model is a representation of the expert's decision-making and information-gathering process. Depending on the task at hand, a variety of entities may be used in inferential flow analysis.

Critical decision method (Klein, 1989b) is a form of retrospective analysis where the expert is presented with critical or unusual incidents and then asked a series of questions designed to elicit factors influencing the decision-making processes. The advantage of this method is its focus on a particular situation and, specifically, on a situation that requires fast or especially

complex inferencing. Such situations often tap the expert's unique knowledge and have the potential of eliciting precisely the type of knowledge that distinguishes expert from simply competent performers. An additional feature of the critical decision method is its use of specialized "probes," that focus on the knowledge type used during problem-solving (e.g., analogies, goals, perceptual cues, other options, etc.). CDM produced lists of critical perceptual cues ("critical cue inventory") and a sequence of situations and associated decision-making parameters (i.e., decision-points, expectations, goals, etc.).

<u>2.3.1.2 Protocol Analysis</u>. Interviewing techniques have the potential disadvantage that the expert may not be reporting what actual thoughts while performing the task, but rather an after the fact reconstruction of the process, which may or may not reflect the *actual* process (Nisbett & Wilson, 1977). One means of avoiding this problem is to ask the expert to perform the task, to "think out loud" while doing so, and then to record the expert's utterances. Protocol analysis is a knowledge acquisition technique that consists of collecting and analyzing the verbal data produced by an expert while performing a task (Ericsson & Simon, 1984). The rationale for this approach is to provide a record of the expert's thought and decision processes as they occur during the actual performance of the task. The experts are instructed to say exactly what is on their mind, as quickly as possible, and not to bother with forming grammatically correct sentences or worrying about being understood, since the record of this process will be available for later analysis. The expert can also be questioned afterwards, to help interpret any remarks that are unclear.

A typical protocol might include phrases such as "let's see, what was I trying to do here?" "oh right! - 7" "this shouldn't be here should it?" etc. , which in and of themselves may not be revealing but in the context of the task, and in conjunction with the entire transcript, often provide important indications about blocks and directions in the experts' inferencing and decision-making processes, as well as reflections of the underlying mental models.

Interruption Analysis is a specialized form of protocol analysis where the expert is interrupted during a critical moment, usually just prior to or just after a decision has been made, and is asked why the particular course of action was selected; i.e., what were all the factors that contributed to that choice. This techniques is used to focus the expert's introspection on a particular segment of the task or a particular aspect of the inferencing process. As is the case with all direct techniques, the risk exists that the reported data may not reflect the actual knowledge or processing taking place.

<u>2.3.2 Indirect Knowledge Elicitation Techniques</u>. Indirect techniques are designed to access implicit knowledge; i.e., knowledge which is difficult to articulate in response to direct questions. As such they are ideally suited to the elicitation of the complex perceptually-derived features that appear to characterize expert problem-solving and decision-making. Each of the techniques identifies the structure and contents of internal representations by eliciting the *classification features* used by the expert to characterize and categorize important *entities* in the domain. Examples of such entities are specific situations on the battlefield, specific configurations of enemy and friendly troops, and specific constraints provided by political, ecological, or logistical circumstances. Examples of attributes are likelihood of success, importance toward overall goal of mission, cost, time required, etc.

Different techniques produce different structures from these elements, including plots in multi-dimensional spaces, various forms of hierarchies, or generalized networks. From the set of available techniques we have selected four to explore in more detail: 1) repertory grid analysis; 2)

multi-dimensional scaling; 3) hierarchical clustering; and 4) the Pathfinder algorithm. These techniques were selected because they represent a well-researched set of methodologies which proved to be effective in the elicitation of implicit knowledge (Berry, 1987; Olson & Biolsi, 1990), and because the statistical software required for their application is readily available.

Repertory grid analysis is a technique adapted from a psychological theory of human cognition, personal constructs theory (PCT) (Kelly, 1955). The central thesis of PCT is that humans organize their world into conceptual entities (i.e., objects or items of interest) and that these entities are classified and differentiated by a series of attributes called "constructs." Judgments and behaviors are the results of manipulating the internal representations of these constructs, either consciously or unconsciously. The technique gets its name from the central structure generated to reflect the experts' constructs; the repertory grid. The grid is a 2-dimensional matrix which classifies the items of interest in terms of a variety of traits and attributes, also termed *constructs*. The basic technique of repertory grid analysis consists of three steps. First, a series of items is selected from the domain. Second, these items are systematically compared by asking the expert to list any similarities and differences they can think of when considering pairs of items. The final step consists of rating each entity along each attribute by assigning a value to the corresponding cell in the repertory grid matrix. Data obtained by repertory grid analysis can be used directly, or can be further analyzed to reveal relationships and structures that might exist among the items (such as causal relationships), or can be source of data for other indirect techniques, such as the proximity scaling techniques described below.

Proximity scaling techniques represent a family of psychometric techniques developed in the 30's and designed to identify internal representations and mental models, particularly where complex, multivariate perceptual data are concerned. The objective of these techniques is to construct a representation of mental models indirectly, from local information about the elements of the domain. The elicitation methodology consists of two steps. First, *local* information is collected in the form of proximity assessments among pairs of domain elements (e.g., specific battlefield situations, possible attack scenarios, etc.). This information is stored in an nxn proximity matrix, where each cell n_{ij} represents the proximity (e.g., similarity or dissimilarity) between items i and j. Different methods exist for eliciting proximity judgments, varying in the directness (i.e., directly asking for similarity judgment or deducing it from other measures) and the time required for data collection. This matrix is then processed by the appropriate algorithm which constructs the corresponding *global* structures, such as maps in some multi-dimensional space, hierarchies, or networks. The structures are then further interpreted, with the help of the expert, to identify salient features, such as the implicit classification dimensions, and complex perceptual features, or, in the case of hierarchies or networks, the meanings of links and substructures. Figure 3 summarizes these techniques.

The rationale behind using proximity assessments as the primary elicited data is that complex perceptual and cognitive judgments are composed of, and therefore can be decomposed into, constituent simpler traits. These techniques are therefore well suited for analyzing complex perceptions into their constituent individual features and have been successfully applied in a variety of areas, including recognition of complex patterns such as those involved in reading x-rays, medical images, and sound analysis.

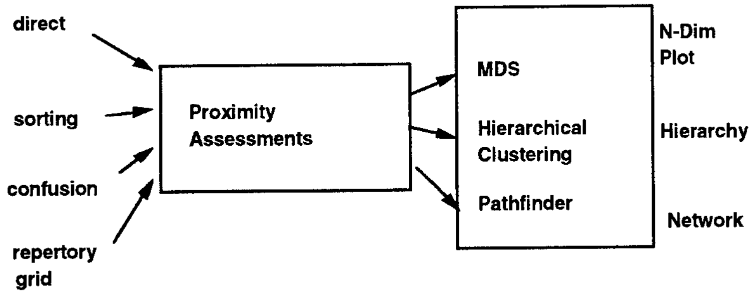


Figure 3. Summary of proximity scaling techniques

Multi-dimensional scaling (MDS) is the most commonly used continuous scaling technique which analyzes complex perceptual data by fitting the elicited proximity data onto a multi-dimensional space, where each dimension represents some global perceptual feature used to categorize the data. These features are identified by asking the experts to interpret the MDS-generated plots. Examples of such dimensions are level of risk vs. cost to human lives, or other complex perceptual features combining information about current context and goals. The key distinguishing features of successful application of MDS are (1) that the dimensions are not directly elicited from the experts and are therefore likely to represent true implicit knowledge, and (2) that these dimensions are independent (orthogonal) and therefore represent a minimal set of dimensions that can be used to classify the input data.

Hierarchical clustering analysis (Johnson, 1967) produces a hierarchical structure of nested clusters of items which correspond to meaningful categories in the expert's mind. The individual clusters can then be further analyzed by the expert knowledge engineer to determine which attributes caused them to be put in the same cluster. Beginning with a proximity matrix, the clustering algorithm iteratively groups items with lowest proximity ratings into clusters, until a complete hierarchy of all items if constructed. Classification dimensions can then be identified from these structures by interpreting the reasons why a set of items is grouped in the same cluster.

Pathfinder algorithm (Cooke, Durso & Schvaneveldt, 1986; Cooke & McDonald, 1988) is a discrete scaling technique which also begins with a proximity matrix and which generates a graph as its output structure. The items in the matrix correspond to individual nodes in the graph and the weighted links correspond to a similarity rating between the items. This initial graph structure is further processed by the Pathfinder algorithm to produce a new network where a "link is present in the output network if and only if that link is a minimum weight path between the nodes connected by the link in the initial (complete) network" (McDonald & Schvaneveldt, 1987). The graphs produced by the Pathfinder algorithm typically contain substructures (groupings of particular items), which can be interpreted by the expert to yield implicit dimensions and traits shared by the elements of such a subgroup.

<u>2.3.3 Elicitation of Multiple Knowledge Structures</u>. A key factor that distinguishes experts from novices is the flexibility and complexity of the internal representations and mental models underlying problem-solving and decisionmaking (Chi, et al., 1988; Klein, 1989b; Cooke & McDonald, 1988). An expert's mental model of a task is more elaborate than the novice's, contains multiple representational structures, and encodes task-specific heuristics that enable the expert to zero-in on the key features of the problem and find an effective solution (Klein,

Calderwood & Macgregor, 1989). Research in human information processing and the nature of expertise has identified a variety of such knowledge structures, which can roughly be divided into those representing declarative (fact-based) knowledge, and procedural (activity-based) knowledge. Declarative knowledge is used to capture factual information about the world, whereas procedural knowledge captures knowledge about acting in the world and enables the expert to perform mental simulation of dynamically-evolving situations.

These internal representational structures include the following:

- *Hierarchical representations* for taxonomies, partonomies, and goal and procedure representation
- *Causal models* for representing knowledge about the domain dynamics and supporting a variety of reasoning including simulation and what-if inferencing, diagnosis, troubleshooting, and planning
- *Direct, analogical representations* such as those required for representing maps
- *Rules* representing heuristics about problem solving and aspects of domain structure

A variety of modifications on these basic themes exist, which generally aggregate the primitives above into larger structures (e.g., MOPS (Schank, 1982).

A critical factor in mental model visualization is therefore the elicitation of these varied knowledge structures, to capture the full complexity of the expert's internal representation and associated inferencing processes. Since no single technique can elicit all of these mental representations, a number of different elicitation techniques is required.

While this fact has been recognized by the knowledge engineering community, and has motivated the emergence of tools and methodologies that employ a repertoire of elicitation techniques rather than focusing on a single one (Cooke, 1994; Leddo & Cohen, 1989), several drawbacks remain:

Emphasis on direct techniques: The majority of existing methodologies focus on the use of direct techniques, which, in addition to the limitations discussed above, have the potential of distorting both the format and the content of elicited knowledge. The human cognitive apparatus is highly flexible and experts are able to tailor their responses to fit the knowledge structures implied by the direct probes. Thus direct techniques can elicit event-based representations when asking a question in terms of events, and state-based representations when framing the question in terms of objects. Little unequivocal empirical evidence exists that these types of direct probes access the presumed structures. The sole use of direct probes, exemplified by techniques such as Cognitive Structure Analysis (Leddo & Cohen, 1989), thus has the potential risk of eliciting the types of structures the interviewer has in mind rather than those actually encoding the expert's knowledge.

Lack of graphical and simulation support for elicitation: While many of the more automated knowledge elicitation tools do include some graphical support, this support tends to be limited to a small number of limited-format abstract displays, such as networks of domain entities or intermediate data formats. The majority of existing tools thus do not provide an environment which supports the knowledge elicitation process by providing a rich graphical representation of the domain-relevant situations or stimuli (e.g., complex graphical representations of battlefield situations), or by allowing the user to construct or modify these graphical displays. There are several consequences of this limitation: *first,* the domain elicitation stimuli must be provided by

the knowledge engineer, which often involves lengthy preparations of drawings and photographs, that may not fully capture the richness and complexity of the task at hand; and *second*, the graphical props used for the elicitation may not be sufficiently realistic to trigger the full extent of the perceptual processes and thus aspects of the relevant knowledge may not be elicited; and *third*, it is difficult or impossible to construct specific scenarios to support case-based elicitation methods.

Limitations of automatic data collection techniques: Non-intrusive observations of expert behavior is an important elicitation technique and can be used in conjunction with both direct and indirect techniques. Such observations are performed while watching the expert perform a task. Most of the available automated elicitation tools are not geared towards such non-intrusive observation, because they do not provide an environment that can adequately support problem-solving or decision-making within the domain of interest. This is in part due to the limitation described above: the lack of adequate case generation facilities, coupled with the lack of rich graphical support tools.

2.4 Display Design Principles and Visualization Technology

2.4.1 Display Dimensions and Design Principles. The multiplicity of mental representations and mental model formats requires a corresponding variety of display and visualization formats. Different formats emphasize different aspects of the task structure or its mental representation. Recent empirical work has identified a number of dimensions humans use to characterize various display formats. These include spatial vs. non-spatial; temporal vs. non-temporal; concrete vs. abstract; continuous vs. discrete; part vs. whole emphasis; depicting a static structure vs. a dynamic process (Lohse, Biolsi, Walkers & Rueter, 1994). These dimensions reflect both the parameters of display designs AND the dimensions along which mental model representations and visualizations can be characterized. They thus outline a space within which to design various display formats and provide guidelines for selecting appropriate formats to match the elicited mental structures.

In addition to the display parameters above, a number of display design principles have been identified that contribute to the effectiveness and comprehensibility of the displays (Larkin & Simon, 1987; Lohse, Biolsi, Walker & Rueter, 1994; Tufte, 1983). Effective displays:

- Take advantage of direct perceptions and minimize the need for complex cognitive computations,
- Combine information about the broader context (e.g., map of the area) with critical relevant task-information about specific situations (e.g., symbolic situation template overlay),
- Minimize the number of steps required to translate the contents of a display into an internal representation by:
    - matching displays to mental structures,
    - emphasizing salient features,
- Maintain a one-to-one mapping between critical conceptual entities and display objects,
- Allow the users to navigate among related formats through links connecting related entities,

- Use "controlled" distortions to emphasize task-relevant information (e.g., fisheye views (Sarkar & Brown (1994)),
- Both reflect and promote internal organization by supplying appropriate abstract structures.

The individual dimension of display formats and the principles above provide guidelines for constructing new displays or modifying existing displays to best capture the elicited mental models.

The state-of-the-art in display generation and multi-media has progressed rapidly in the last five years. As is the case with decision-systems, the technological advances have outpaced theoretical display design methods. However, both technological and theoretical progress has been made resulting in a wide variety of media available on high-resolution devices supporting sophisticated visualization and display techniques. From a theoretical point of view we are just beginning to understand the human element in display design: what types of information are best displayed in what format (Lewis & Fallesen (1988)) and, specifically, how information elements should be combined in a battlefield setting to best communicate tactically-relevant knowledge (Badre (1978)). Empirical studies are being conducted to understand the dimensions of displays (e.g., spatial vs. non-spatial; concrete vs. abstract, static vs. dynamic, temporal vs. non-temporal, etc.) (Lohse et al. (1994)).

Such systematic classification lays the groundwork for a systematic design of display formats that best captures the individual and task requirements. The technology enables the generation of unusual displays, such as distorted maps and displays called fisheye views, which help emphasize a particular aspect of a pictorial representation to convey some information (Sarkar & Brown (1994)). For example, a fisheye view of a COA action-reaction tree might encode some parameter (e.g., level of risk) in terms of the size of the various nodes in the tree. Such a display would enable the commander to instantly recognize situations of high risk. The available technology and data gathering systems (e.g., ummanned aerial vehicles, GPS tracking, etc.) support novel combinations of battlefield information and move towards a true virtual reality planning environment. For example, integration of 3-D visual simulations and video imaging can support very realistic and perceptually compelling war-gaming (Witte & Kelly, 1994; Walter & Warren, 1992).

2.4.2 Visualization Technology. A review of visualization software for implementing the VIEW prototype focused on three operating systems: Unix/X-Windows, Macintosh OS, and DOS/Windows. Although exceptionally good graphics capabilities are supported by Unix machines, such as the Silicon Graphics Inc. Iris series, the relatively high price/performance ratios eliminated them from further consideration as potential hosts in what could eventually grow to be a large network of low-cost hosts. We thus favored the Macintosh OS and DOS/Windows environments. Although the former provides superior graphics tools, we selected the latter because of the much larger installed base at ARI, and the greater likelihood of integration/networking with existing Army systems.

With the focus on DOS/Windows-based software, we quickly identified four key software packages: Visual Basic for the interface, Microsoft Access for the database, Visio Technical for graphical support, and CLIPS for ruleset implementation. These applications all support Dynamic Data Exchange (DDE), so that, the applications can be easily linked together. Since Windows is a multitasking system, many event-driven programs or applications are permitted to run concurrently. The DDE feature of Windows allows an application to directly and continuously exchange data with other Window-based applications that support DDE.

Visual Basic is an object-oriented, Window-based programming language that facilitates the utilization of objects to initiate the execution of different programs and applications. It can be used for implementing both the tactical visualization interface and the object world model, to be described in the next chapter. Via DDE, Visual Basic can be the environment through which objects developed or residing in other applications are linked or embedded. Visual Basic allows for the development of interactive user interfaces and is ideal for accessing and manipulating databases because its objects have a set of assigned attributes: properties and methods. These attributes define how and when an object will respond to events. Database objects in Visual Basic, which are logical representations of physical databases, have properties and methods that can be used to manage data. Interface objects provide *smart* controls, with their customized attributes, to be integrated in stand-alone applications.

Visual Basic uses the Microsoft Access database engine for its local data update and retrieval functionality; this can be used to implement the object database, to be described in the next chapter. This permits stand-alone executables made in Visual Basic that manipulate databases developed in Access and graphical icons developed in other graphics applications such as Visio Technical.

Visio Technical is a software package designed to run with Microsoft applications, and can be used to support development of the graphical interface. Visio allows the creation of unique icons and other graphics that can pictorially encode many forms of information. Since people often process pictures faster than words, pictorially encoded information is ideal for expressing ideas. When used in conjunction with a Visual Basic application, Visio provides tools necessary for a custom designed interface that facilitates the displaying and retrieval of information.

CLIPS is software developed at NASA's Johnson Space Center, and can be used for implementing any formal ruleset. It provides a rule/object-based environment in which to develop an expert system. It is ideal for the development of situation-driven queries and situation-specific visualizations, because it facilitates the structuring of information into logical segments. Additionally, it is DDE compatible so it can be linked to the other applications proposed here.

3. System Description

This chapter provides a functional description of the VIEW prototype. Section 3.1 defines the system architecture and provides a general overview of system functionality. Sections 3.2 and 3.3 then describe the two key subsystems, the Visualization Subsystem and the Knowledge Elicitation Subsystem, respectively.

3.1 System Architecture and Functional Overview

The VIEW prototype provides a range of functionalities for the commander and the knowledge engineer. The architecture consists of two distinct but tightly coupled subsystems: the Visualization Subsystem and the Knowledge Elicitation Subsystem. The Visualization Subsystem is used to construct, store, browse and view a variety of displays depicting different aspects of the battlefield situation. The Knowledge Elicitation Subsystem uses the full functionality of the Visualization Subsystem and adds to this the capability to elicit specific types of data from the commander using a number of KE techniques.

Figure 4 defines the VIEW system architecture. As shown, it is composed of two major subsystems: the Visualization Subsystem, and the Knowledge Elicitation (KE) Subsystem.

The Visualization Subsystem shown in the figure is composed of three interlinked modules: the Tactical Visualization Interface, the Object Database, and the Object World Model.

The Tactical Visualization Interface supports the commander in two basic ways. First, it provides him with situation-relevant tactical information, displayed in a manner best suited for his analysis/planning tasks, and best matched to his individual preferences. Second, it provides him with the means of directly manipulating the object database, to create or modify the tactical situation, to explore alternatives, to conduct simplified wargaming exercises, and the like. As shown in the diagram, a graphical user interface supports navigation across a range of displays maintained in a display library.

The Object Database provides a common object representation for all visualization/elicitation component of the system, and is directly linked to the Tactical Visualization Interface via tactical commands generated by the user and object attributes sent to the displays. Three general classes of objects are maintained in the object class library: 1) terrain-related objects (terrain elevations, vegetation, roads, etc.; 2) military unit objects (echelons, types, weapons systems, etc.); and 3) ground environment objects (battlefield AO/AI, avenues of advance/approach, etc.). As shown in the diagram, the object database also serves the KE Subsystem by providing it with "snapshots" of the current tactical situation.

The Object World Model supports an object-oriented simulation of both friendly and enemy forces operating over a specified battlefield reflecting weather and other environmental conditions. Linked to the Object Database via object commands and states, the module provides a direct means of dynamically modifying the database over time. An object behavior library supports the simulations of friendly/enemy mobility, and, via extension, wargaming capabilities.

The Knowledge Elicitation Subsystem shown in the figure is composed of two interlinked modules: the KE interface, and the KE Recording/Analysis Module.

The KE Interface supports the knowledge engineer in three ways. First, it provides a means of navigating among the KE techniques, via the control interface. Second, it supports the collection of elicited data from the commander who is interacting with the Visualization Subsystem. Finally, it provides on-line access to the results of KE analysis, to support interactive navigation amongst the displays, as a function of the results of the analysis. As shown in the diagram, a graphical user interface supports navigation across a range of techniques maintained in a KE library.

The KE Recording/Analysis Module implements the actual recording and analysis of the elicited data via direct links to the KE Interface. In addition, to insure close linkage with the Visualization Subsystem, the recording module also accepts as inputs the Visualization configurations selected by the commander via the Tactical Visualization Interface, as well as "snapshots" of the tactical situation as maintained by the Object World Model.

The demonstration software was implemented as a Visio Technical extension. The Visio extension approach to software development involved three interrelated steps.

The first step in the development process involved creating and saving a specific multi-window Visio workspace by modifying the Visio development environment to the specific requirements of this application. The term workspace here refers to a collection of interactive interfaces that are integrated based on a specific design and hierarchy.

The second step consisted of adding functionality to the software and its host environment (i.e. the workspace) by embedding stand-alone and functionally independent executables in the environment itself. The stand-alone executables were developed in the Visual Basic development environment. This Windows-based package is very suitable for fast

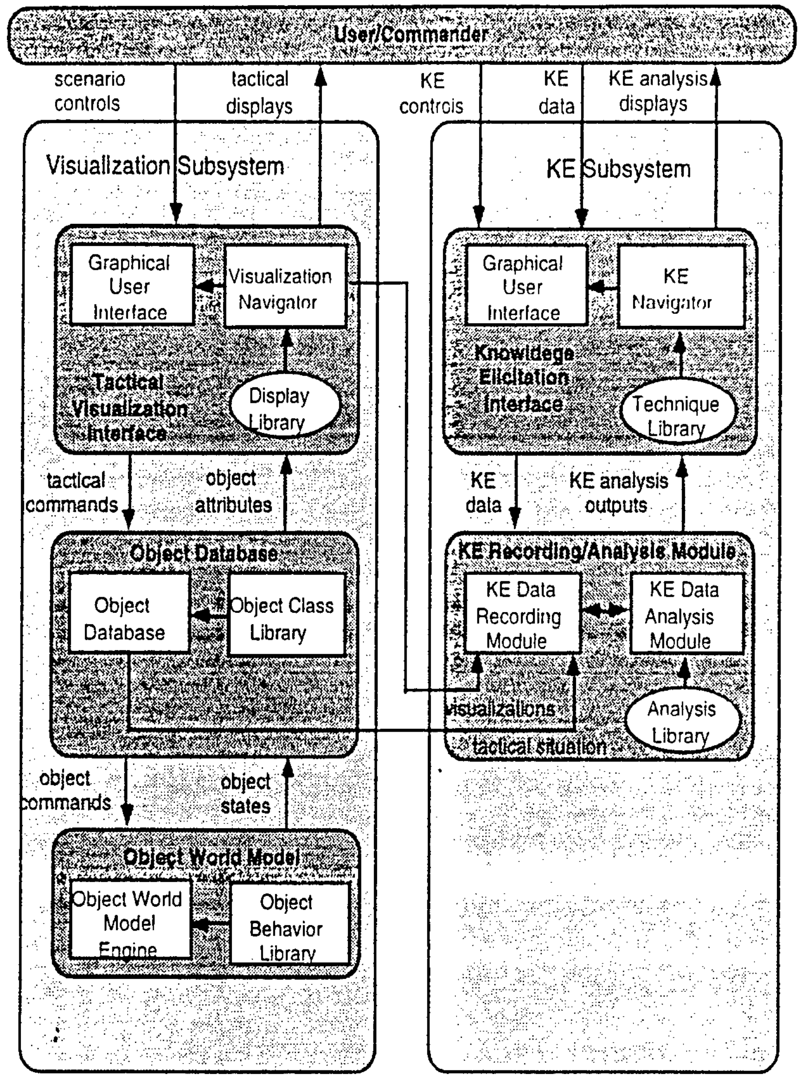


Figure 4. Block diagram of system architecture

implementation of software designs that involve multiple interrelated interfaces. Furthermore, this development language has provisions for fast and easy access to databases created in the Microsoft Access environment. In addition to linking all objects in the workspace to the Microsoft Access databases, using Visual Basic for developing the executables has also rendered the overall environment more flexible for the user.

The third and final step in the development process involved adding functionality to various objects in the workspace by building stand-alone Visual Basic executables. These executables perform several types of tasks depending on the nature of the object they are linked to. For example, these executables give user access to different interfaces, as well as object attributes that would be otherwise hidden from the user. Through these executables, object databases are updated whenever the user modifies an object attribute through any of the interactive interfaces.

We now describe the major subsystems of the VIEW prototype.

3.2 Visualization Subsystem

The Visualization Subsystem provides the user with a library of display formats, which can be customized to fit the current situation (e.g., units can be placed on a terrain to depict current plan of attack). The displays can be combined to form distinct user interface configurations and the user can navigate among the different display types as s/he is examining the battlefield situation making tactical decisions. As illustrated in figure 5, the system captures workspace flexibility in adjusting to user preferences.

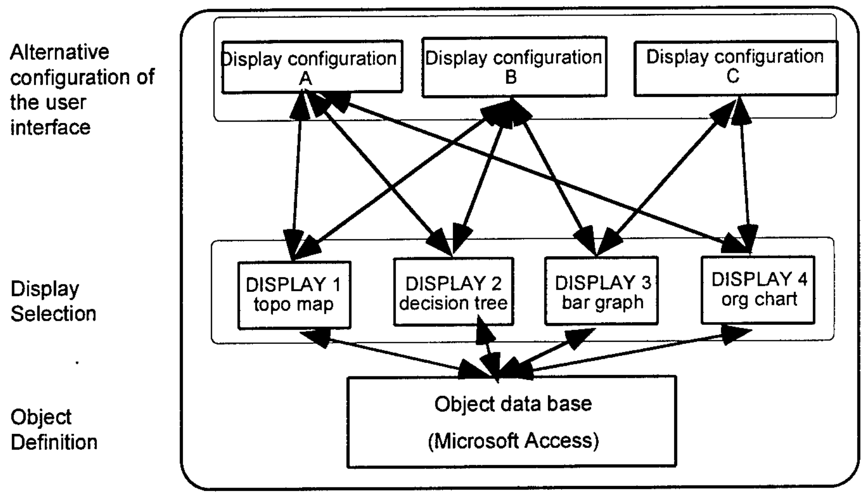


Figure 5. Illustration of system use to create a variety of visualizations

All domain knowledge is encoded in objects and their attributes, which are stored in a Microsoft Access database. This object-definition layer thus forms the knowledge-base from which all displayed information is derived. When an object is defined or manipulated by a user on the screen, the relevant information is defined or modified in the database. For example, if the user moves a battalion symbol on the map, the new position of the symbol is updated in the database.

Figure 6 shows the overall architecture of the Visualization Subsystem developed for the Phase I prototype. There are three main modules that define system functionality. These are the Tactical Visualization Interface Module, the Database Module, and the Queries/Rules Module.

The functional relationship between these three modules is shown in figure 6. The user manipulates objects on the Tactical Visualization Interface by changing or modifying object attributes on the screen through the scenario controls. The user can also directly access the Queries/Rules module to inquire about specific information regarding object attribute relationships in order to get information regarding the dynamic battlefield situation. All actions by the user are recorded in the database module. By keeping the object database current, the system performs on-line queries that are based on current object attributes and relationships.

In addition, the user can use the Visualization Navigator directly from the Visio workspace to display different attributes of the same object simultaneously on independent displays. These displays are supplied to the workspace from the Display Library.

In the following sections, description of each main module and examples of the corresponding interfaces will be given.

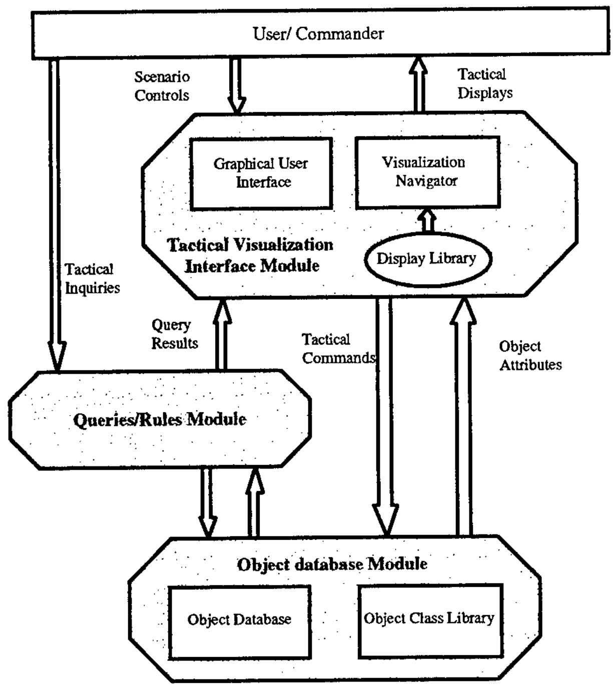


Figure 6. Block diagram of visualization subsystem architecture

<u>3.2.1 Tactical Visualization Interface Module</u>.

This module is composed of three main components that are functionally independent and are directly accessible from the Visio environment.

Graphical User Interface: This component of the Tactical Visualization Interface is directly provided by the Visio workspace. Figure 7 shows an instance of this interface.

Visualization Navigator: Figure 8 shows the Visualization Navigator interface as seen by the user in the Visio workspace. The navigator allows the user to choose an object class and open multiple windows depicting different attributes of the same object in different display formats. Three visualization classes are provided for: Units, Map Display, and Scenario.

For example, clicking the "Units" button will open up a dialog box with a collection of options to choose from. This dialog box represents the top level Units display panel, shown in figure 9, and provides access to all available visualizations of unit attributes. The selected unit designation is shown on the top left of the screen.

Each checkbox represents a set of related information organized together. For example, figure 10 shows how clicking on the Logistics checkbox opens a new interface with supply levels for all 10 classes of logistical supplies. The logistics information depicted on the bar graph is unique to the selected unit and is directly read from the Microsoft Access database.

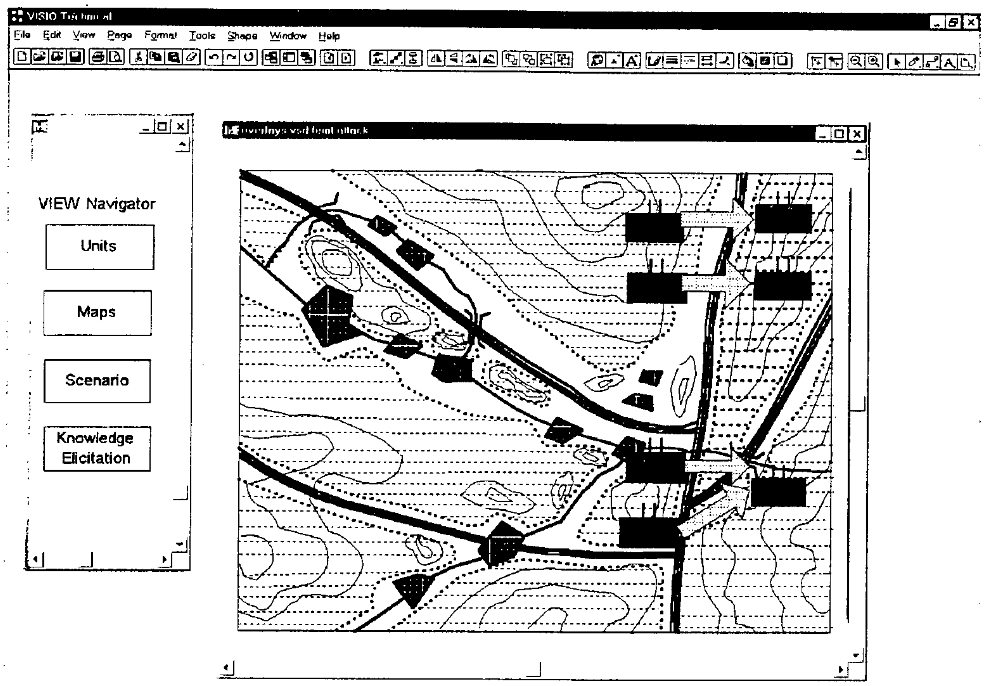


Figure 7. Visio workspace & tactical visualization interface

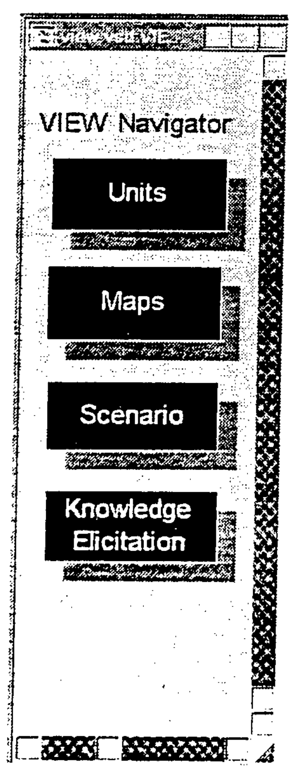


Figure 8. Visualization navigator interface

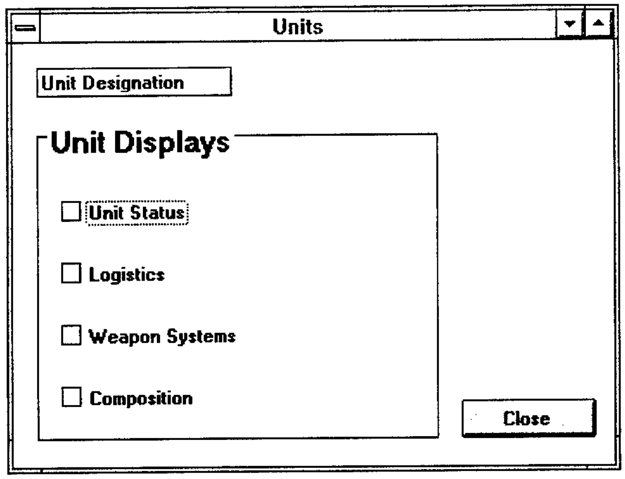


Figure 9. Top level display panel for units display

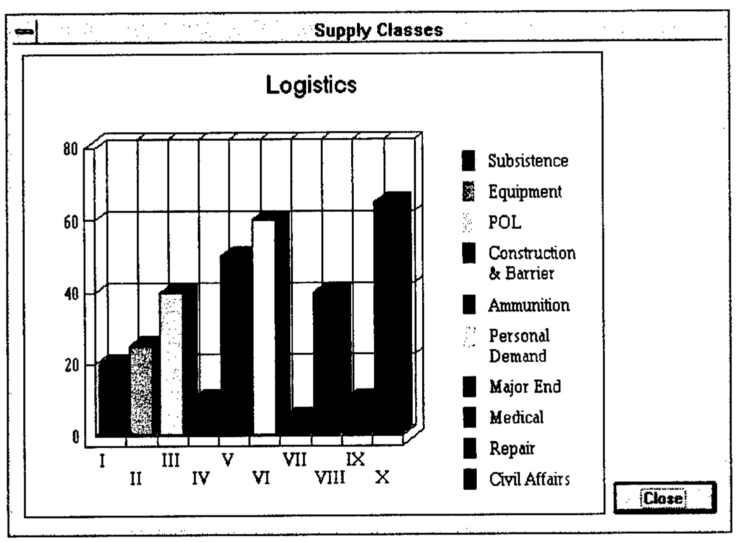


Figure 10. Unit logistics: supply classes

Display Library: To provide a flexible visualization environment for the battlefield commander the display library of the VIEW prototype workstation contains a variety of basic display formats which can be modified and customized to reflect the current situation or the commander's preferences. Each display emphasizes a different combination of the empirically-derived display/mental model parameters and thus different displays are suited for different types of inferencing and information integration. The display formats are shown in figures 11 through 18 and described in the text below.

In developing the Display Library for the Visualization Subsystem, Visio environment capabilities were complemented with interface design capabilities of other Windows-based applications such as Microsoft Visual Basic (combining checkboxes, radio buttons, etc.) and Microsoft Excel (spreadsheets). These applications were accessed through the Object Linking and Embedding (OLE) capability of the Windows environment. The types of displays used for the demonstration software include bar graph, table, Windows dialog boxes, and different types of military-specific templates and overlays. The display library of the prototype workspace consists of the following visualizations.

• *Dialog Boxes:* Figure 11 shows one of the many dialog boxes provided for the user in the prototype software. Dialog boxes provide one form of user interface where the user can input choices. The example in Figure 11 shows the main menu for friendly forces task organization.

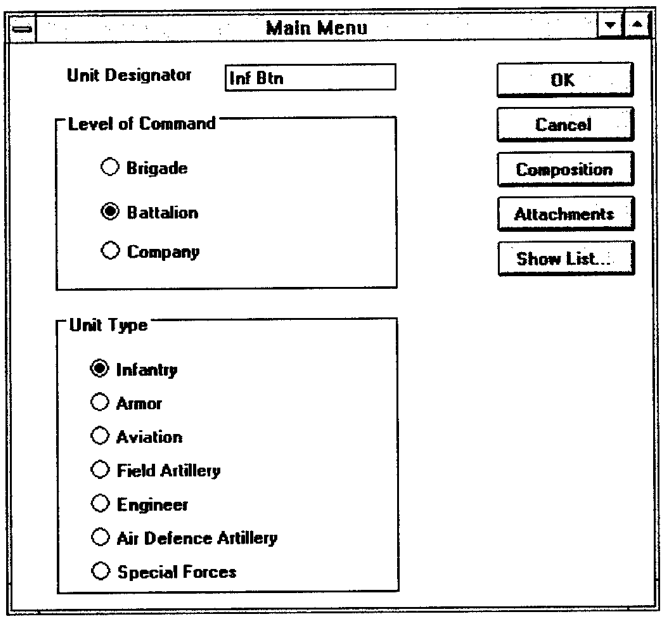


Figure 11. Example dialog box in display library

• *Text Display:* While pictures are generally worth a thousand words, occasionally a textual format is the best way to quickly communicate specific information or information for which there are no simple, generally understandable graphical displays. The VIEW prototype workstation allows for such display of *free text* as shown in the example display depicted in Figure 12.

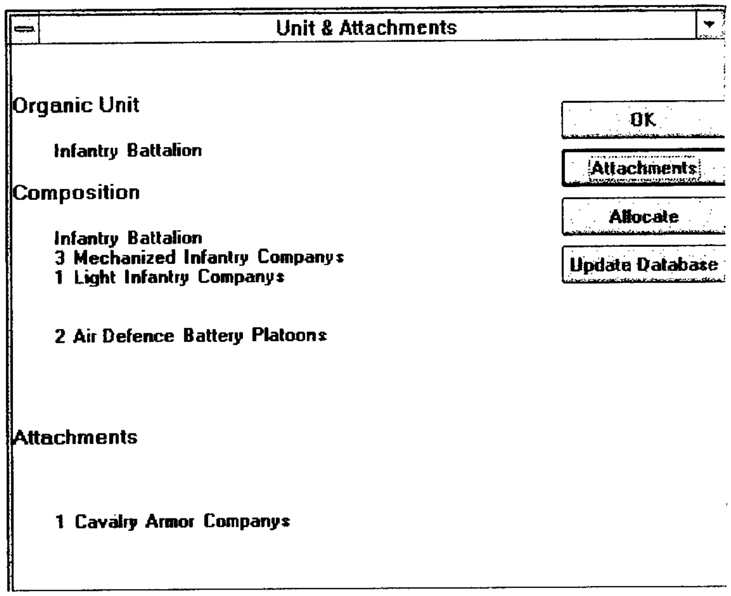


Figure 12. Example of text display in display library

• *Bar Graph:* The familiar bar graph is an efficient and effective means of rapidly displaying the same type of information (e.g., remaining or required quantity) about a number of different variables (e.g., different resources). This display is ideally suited for depicting the resource and logistical requirements at different levels of resolution (e.g., different units, different COA's, etc.) The format of the display lends itself to a fast assimilation of the relative status of a large number of variables.

Anomalies can be identified quickly and in a single scan. A variety of enhancements are possible, including indications of levels of certainty about the information (by different color coding), and indications of limits and desirable ranges on the specific variables (e.g., show required, actual, and available level of resources). Figure 13 shows an example of Bar Graph used in the Visio workspace.

• *Table/Spreadsheet:* Figure 14 shows an example of a table display which is available in the Display Library. The example shows the Weapon Systems for a tank battalion.

• *Maps & Overlays:* Current Army training and visualization tools rely heavily on the use of maps and overlays that emphasize a particular aspect of the task domain. These combined displays have a number of advantages: they represent a large amount of information in a readily understandable, familiar format; they combine spatial representations (which trigger lower-level perceptual processing) with abstract symbology (which trigger higher-level symbolic processing), thus providing both an overall context (e.g., map of an entire area) and a specific aspect of the situation on which to focus (e.g., arrows representing movement, icons representing units and weapons; etc.). With appropriate overlays these maps can also capture the time dimensions of the problem, such as in the case of decision support templates.

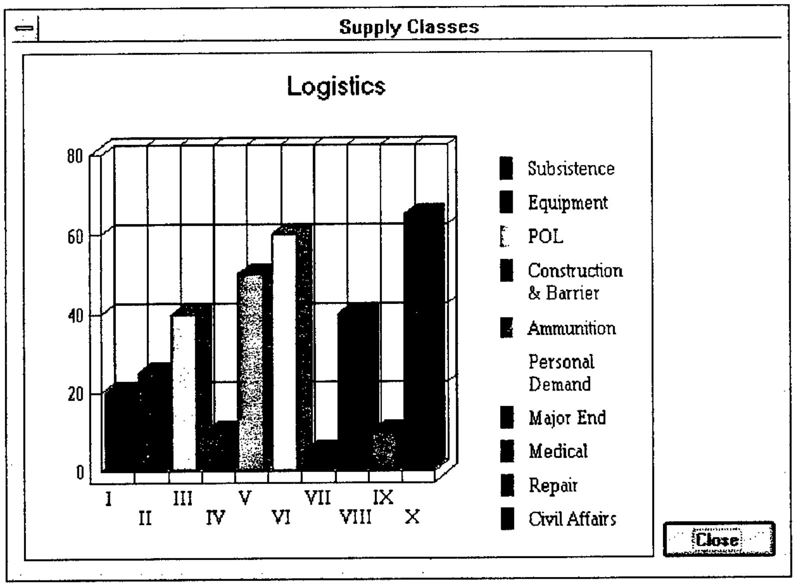


Figure 13. Example of bar graph in display library

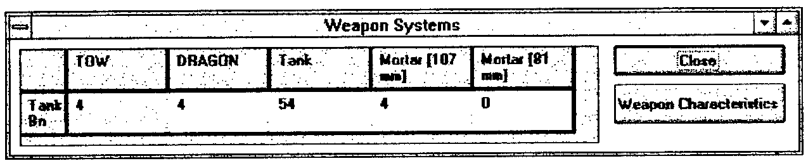


Figure 14. Example of table display in display library

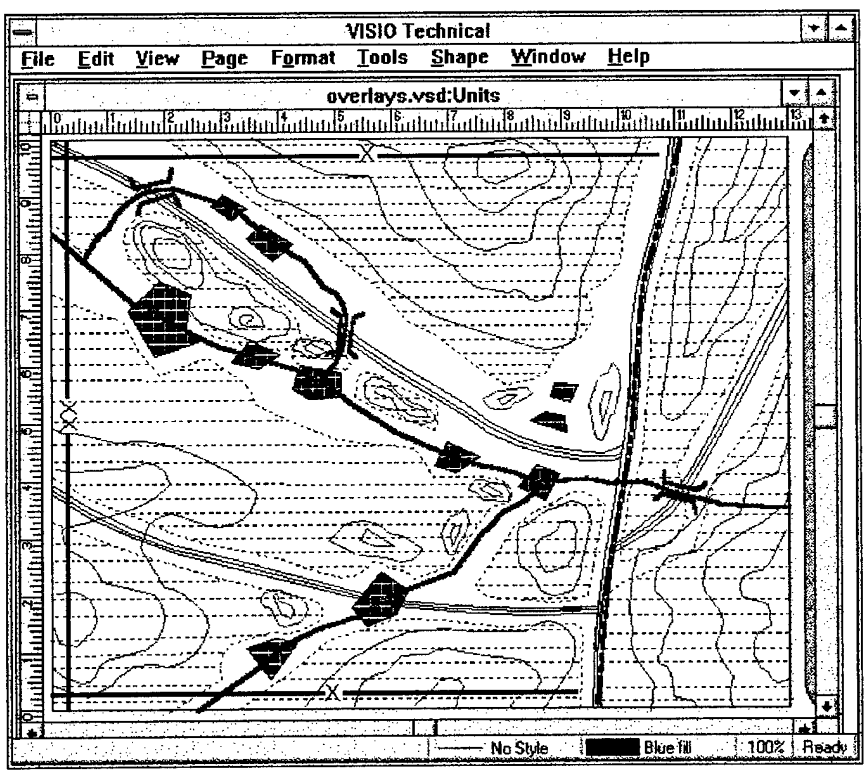


Figure 15. Example of map display in display library

Figure 15 shows an overlayed collection of maps displayed in the Visio environment. The overlays include cities, roads, vegetation, terrain elevation, contours, and rivers.

• *Organizational Charts:* While new display formats can capture a unique way of viewing information, in many cases an enhancement of an existing display format is sufficient to create a powerful means of filtering and combining relevant information. A hierarchical depiction of the unit composition is an example of such a display format. The familiar hierarchy provides an overall context, allowing the commander to view units at different levels of hierarchy in the same *scan*, and providing a display background on which a variety of information (i.e. different characteristics of the particular unit, weapons, resources available, level of combat readiness, etc.) can be overlayed.

Figure 16 shows an example of an Organizational Chart available in the Display Library. The present chart depicts the composition for an Infantry Brigade.

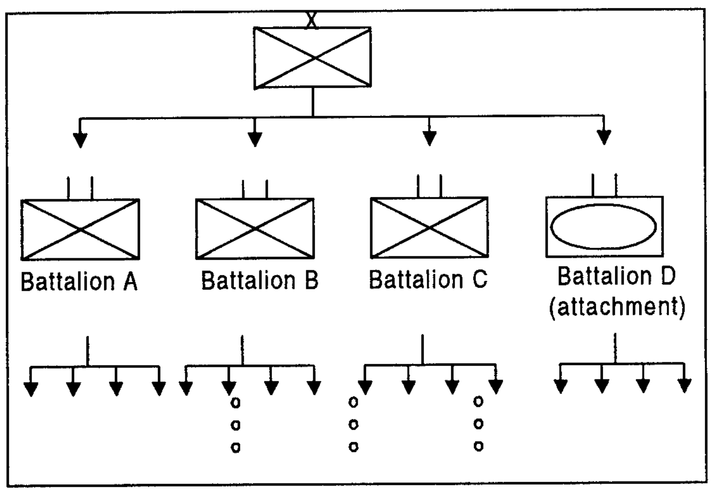


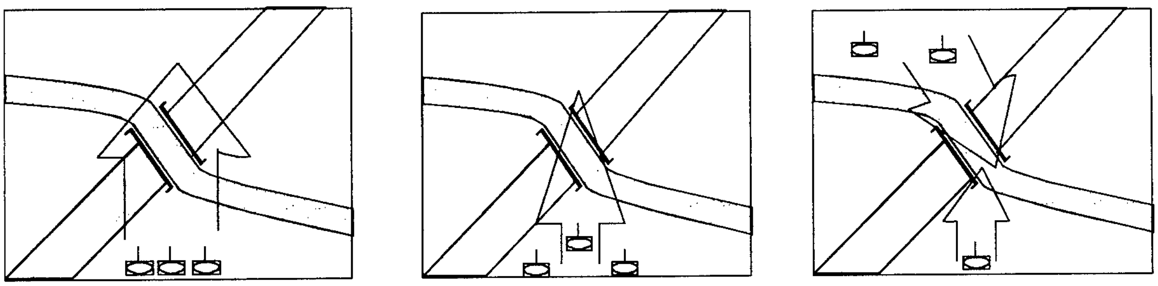
Figure 16. Example of organizational chart in display library

• *Decision Tree:* Another hierarchical display, the decision tree, is unique in that it combines a trace of a cognitive process over time; namely, it provides a trace of the decision making process with respect to the development of a particular COA sequence. Thus, time is an implicit dimension in this display. Furthermore, the display is highly abstract and symbolic, depicting a series of complex situations by a single labeled node in a tree diagram. As such, this display is well suited as a type of navigation backbone through which access to the variety of other displays and information about the situation is available.

Figure 17 depicts a sample decisions, with three COAs presented to the user both in graphical, as well as textual form.

Mission for Battalion D

COA 1 | COA 2 | COA3



all 3 companies attack bridge from the south

one company attacks from the south and two follow

one company attacks from the south, one company attacks from the north with the third company following

Figure 17. Example of decisions in display library

• *Synchronization Matrix:* The familiar synchronization matrix is another display with a time dimension. In terms of information content, it is similar to the decision support template. However, its tabular display format makes multiple types of information readily visible. By making explicit the activities of the different units across time, the synchronization matrix allows the commander to quickly determine whether conflicts exist between these activities or whether there are gaps in the sequence of planned actions.

Figure 18 shows an example of a Synchronization Matrix available in the Display Library.

H-3.5 H-3 H-2 H-1 H.05 H

Military Police	secure western bridges	control western bridges			
Battalion A			leave assembly area	engineers prepare river crossing	
Battalion B			leave assembly area	engineers prepare river crossing	
Battalion C			leave assembly area		engineers prepare river crossing
Battalion D			leave assembly area		move to assault area

Figure 18. Example of synchronization matrix in display library

3.2.2 Object Database Module.

All objects are stored in a Microsoft Access database. The functionality of the Microsoft Access application allows for the creation of relational databases, so that the same object can be included in many tables. Additionally, the flexibility of the application allows for the creation of new fields and records on an as-needed basis to facilitate the creation of new objects and new attributes "on-the-fly" during visualization/elicitation session.

The inclusion of the database in the overall structure has made it possible to decouple the object-definition layer from the display definition layer. Therefore, a given object may exist in several displays and, conversely, a number of display objects may exist for a given object. Hence, there exists a one-to-many mapping between the object-definition layer and the display-definition layer. For example, a particular battalion may be part of several situation templates, several decision support templates, and an organizational chart.

Some of the objects correspond to actual physical objects (e.g., rivers, roads, cities); others correspond to conceptual entities in the Army domain (e.g., area of interest, mobility corridors, avenues of advance/approach); and others correspond to useful abstractions (e.g., natural objects, man-made objects). All objects share the same hierarchical description structure:

object

attribute

attribute of attributes...

The natural object, terrain, has many attributes: elevation, slope, vegetation, soil type, etc. Therefore, the database contains a table for the terrain and many fields for its attributes. For each attribute such as vegetation, there exist other attributes such as vegetation type, vegetation density, and vegetation height. Again, for each attribute there exists a table with many fields pertaining to attributes of attributes.

The Army domain object of a brigade has many attributes, both qualitative and quantitative in nature. First, there are attributes which are qualitative descriptors of the brigade such as strength, training quality, and activity level. Second, there are attributes that describe the brigade in a more quantitative sense such as its composition and attachments. For example, a brigade contains battalions which are made up of companies which are comprised of platoons. All of these attributes, which are also objects, can be organized in a database. From this hierarchical structure, one can see the importance of using a relational database in the design of the software.

3.2.3 Queries/Rules Module.

A critical component of the VIEW prototype is the support it provides for automatic detection of specific conditions of the terrain, units, resources, or overall situation that might be of interest during planning. These conditions are expressed either as queries to the system, as rules defining some alarm or alert condition, or as a general situation of interest.

Queries and rules are used to represent situations that might be desirable or undesirable and are a means of automatically detecting particular situations and displaying relevant information to the commander. Queries and rules thus serve the function of an *intelligent assistant*, who is aware of particular conditions which the commander should be cognizant of and notifies the commander when conditions occur. Examples of alarms or alerts are:

- Reaching a particular level of resources supply (fuel, food, ammunition),
- River current above x,
- River depth below x or above y,

- River banks with slope < x or more than y,
- River banks with or without vegetation,
- Areas above height x (for artillery, communication placement),
- Areas with:

  no vegetation cover,
  vegetation above height x and below density y,
  vegetation above density y,
- Hills with slope < x and slope > y.

The Queries/Rules module was created and integrated within the overall system architecture to give the user the above functionality in an accessible manner. The object database, and the capability of the system to update it on-line provides the infrastructure for developing this module. This module bases its functionality on two general sets of object attribute relationships:

Single-Object Attribute Relationships: These types of queries/rules occur when the question formed by the user presupposes attributes of a single object. An example of this case occurs when the question is about a combination of attributes of a river object: "show all river locations where Current=slow AND Width>50 ft"; Another example is when the user inquires about all units where "Attachment includes an Engineering Company AND Maintenance Supply=Low".

Multiple-Object Attribute Relationships: These types of queries/rules occur when the user inquires about relationships among objects. In this case for the module to answer the question, it should have access to the most recent attributes of multiple objects. An example of this case is when the answer to the question requires information about attributes of different objects. For example, "If Unit Type=Armor AND Disposition Elevation>300 ft", Notify Commander", is a type of rule where terrain object attributes and unit attributes are simultaneously needed for an answer to be produced. Part of the queries/rules interface is shown in figure 19. This interface allows the user to 1) load a query that was previously saved, or 2) create a new query. Figure 20 shows the interface for forming a query/rule. In this example, the user is interested in vegetation type and area elevation and uses the pull down menus to form a compound Boolean statement. The user then selects a region of interest which to query as in figure 21 and then executes the query. The results are shown in figure 22.

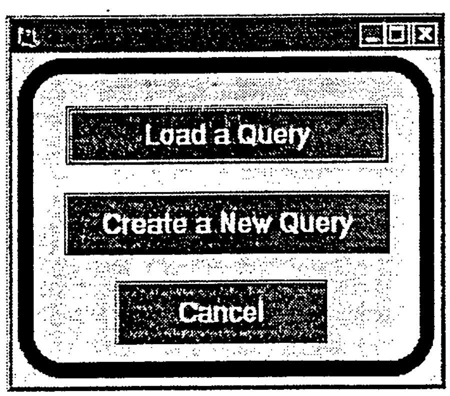


Figure 19. User interface for selecting a query

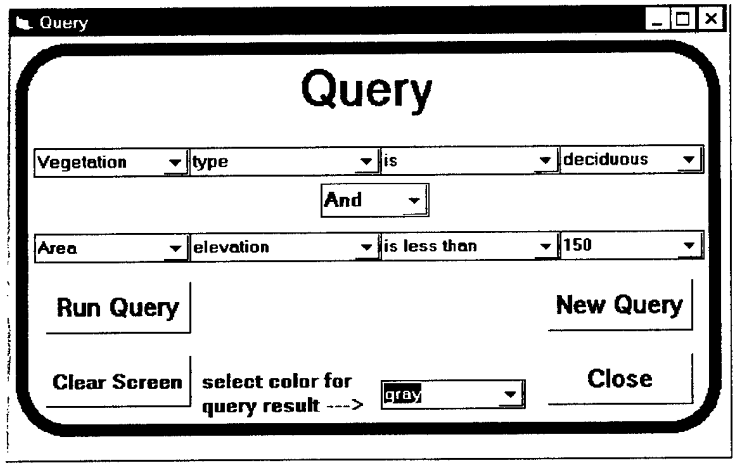


Figure 20. User interface for forming a query

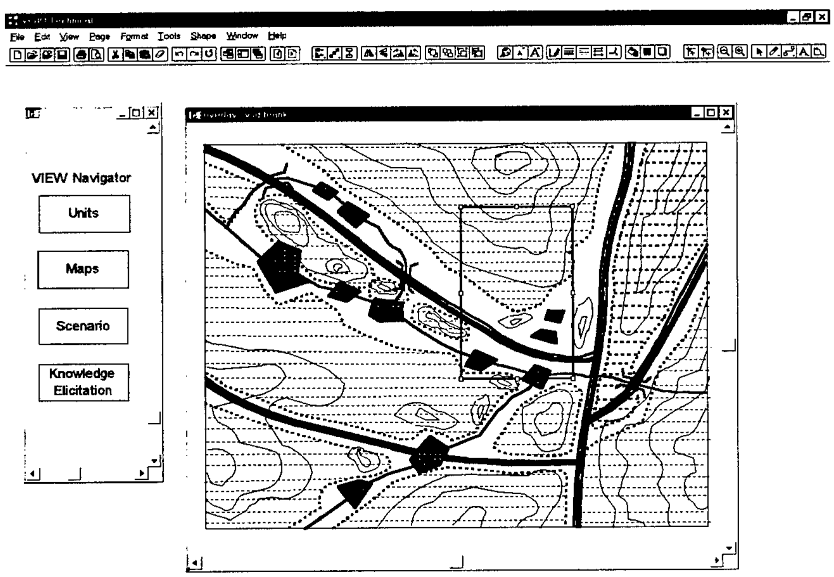


Figure 21. Region selected to be queried

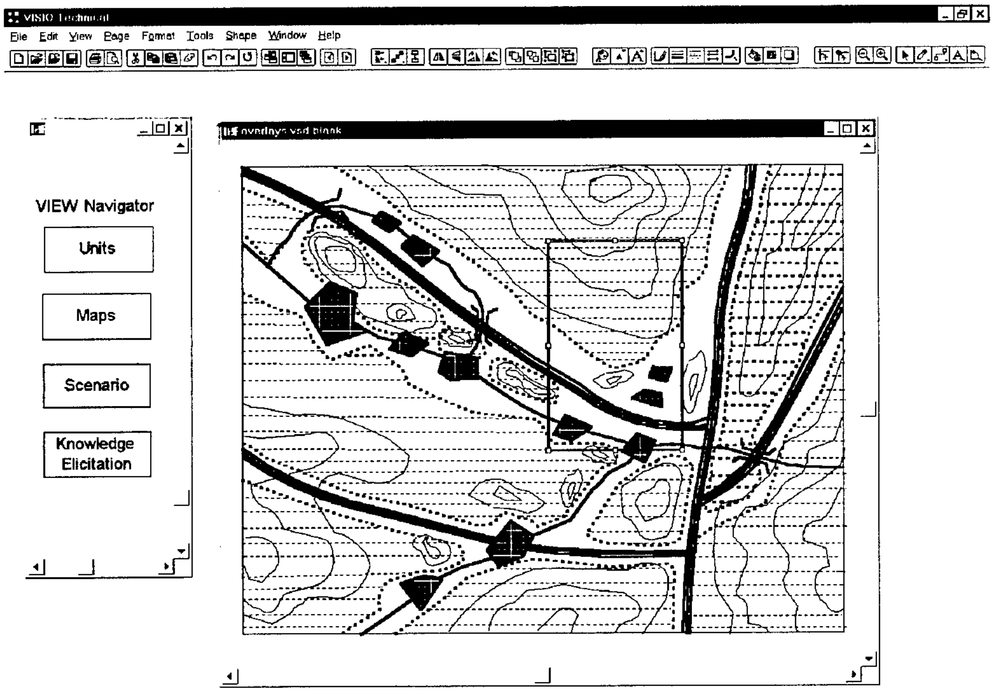


Figure 22. Results of query

3.3 Knowledge Elicitation Subsystem

The design of the VIEW prototype provides the knowledge engineer (KE) with a wide variety of tools to support the process of knowledge elicitation, the subsequent data analysis, and the final interpretation of the results. The workstation design provides an environment within which a variety of knowledge elicitation techniques can be performed, both direct and indirect, and a variety of data collection methods can be employed to support these techniques. The knowledge elicitation component of the design is tightly coupled with the visualization component, and thus the full-functionality of the visualization component is available to the knowledge engineer and the subject matter expert (SME). The prototype design includes the following functionalities:

Graphical Displays and Visualizations:

- A library of graphical displays which can be used either as a basis for direct elicitation techniques (e.g., depiction of a particular situation in critical decision method analysis) or as stimuli in indirect elicitation (e.g., entities for comparison in repertory grid analysis or proximity scaling techniques) As described on the previous section, these displays exist at varying levels of complexity, ranging from simple icons to complex situation and doctrinal templates.

- Support for a variety of data collection techniques where different displays and stimuli are systematically presented to the SME to elicit qualitative judgments for direct elicitation techniques and ratings and assessments for indirect techniques (e.g., pairs of situation templates depicting different COAs can be presented to elicit proximity judgments for multi-dimensional scaling or hierarchical clustering).

Direct Elicitation Techniques:

- Facilities for entering and analyzing free-form text while viewing different displays for a particular scenario,
- Facilities for constructing and editing domain vocabularies and concept maps during the elicitation session,
- Facilities for constructing aggregate structures from these domain primitives to reflect the SME's mental models,
- Facilities for editing and browsing the elicited structures.

Indirect Elicitation Techniques:

- Facilities for editing and transformation of the elicited data (e.g., data from multiple experts can be combined and data in one format (e.g., repertory grid) can be transformed into another format (e.g., proximity matrix)),
- A repertoire of statistical techniques for analyzing the elicited data, ranging from simple correlations and factor analysis, to multi-dimensional scaling and hierarchical cluster analysis,
- A flexible environment for displaying the analyzed data and for assisting with the interpretation process (e.g., displaying MDS plots from different perspectives and providing 2-dimensional projections of complex spaces to facilitate analysis).

In combination, these functionalities provide a dynamic, flexible environment within which a variety of knowledge elicitation techniques can be experimented with and used for different purposes.

Three features of the VIEW prototype design are particularly critical for the knowledge elicitation process and distinguish VIEW from most existing automated knowledge elicitation tools (Bradshaw, Ford, Adams-Webster & Boose, 1993):

- The ability to construct and use a wide variety of graphical displays as stimuli for the knowledge elicitation process. Such depictions of the actual situations are more effective at triggering the perceptual and cognitive processes of the SME, thereby generating more accurate reflection of the SME's knowledge.
- The ability to track the SME's use of the rich set of available displays while solving a particular problem, thereby collecting non-intrusive data about the use of the various display formats. Inferences can be made from these data about the nature of the underlying mental representations activated and used during situation assessment decisionmaking. This methodology is consistent with recent work in constructivist approaches to knowledge elicitation (Bradshaw et al., 1993).
- The ability to integrate support for both direct and indirect knowledge elicitation techniques. For *direct techniques*, which typically involve qualitative rather than quantitative analysis, the support consists of: 1) providing a rich set of graphical displays which can be edited to

suit the purposes of the specific elicitation session; 2) providing customized *probes* for eliciting particular aspects of a situation (e.g., systematically asking the expert specific sets of questions); 3) collecting the elicited information on-line; and 4) displaying, where appropriate, emerging abstract structures (e.g., semantic nets, hierarchies, conceptual graphs, domain vocabularies, etc.). For *indirect techniques*, which require more complex data collection and sophisticated analytical techniques, the VIEW design provides for collecting in one place all the required tools and provides facilities for creating and displaying complex and realistic graphical elicitation stimuli.

Following the general architecture of the KE Subsystem illustrated in figure 4 earlier, we now describe the key modules in greater detail.

3.3.1 Knowledge Elicitation Interface.

The user (an SME or a KE) interacts with the workstation through a menu-driven graphical user interface through which choices are made about which technique to use, what method to use for data collection, and how to process the data and interpret the results. Figures 23 and 24 illustrate two snapshots of the elicitation interface as it might be used during a particular indirect KE session. The selections made by the user through this GUI are recorded in the system, so that at any point in time the system knows which technique is being used, which data collection techniques are therefore applicable, and the status of the intermediate data formats. The user can move back and forth between techniques, trying out different combinations as required by the task, and in general, experimenting with different mixes of data collection and analysis methods. The VIEW prototype design thus supports the entire elicitation process, beginning with data collection and ending with the analysis of the results.

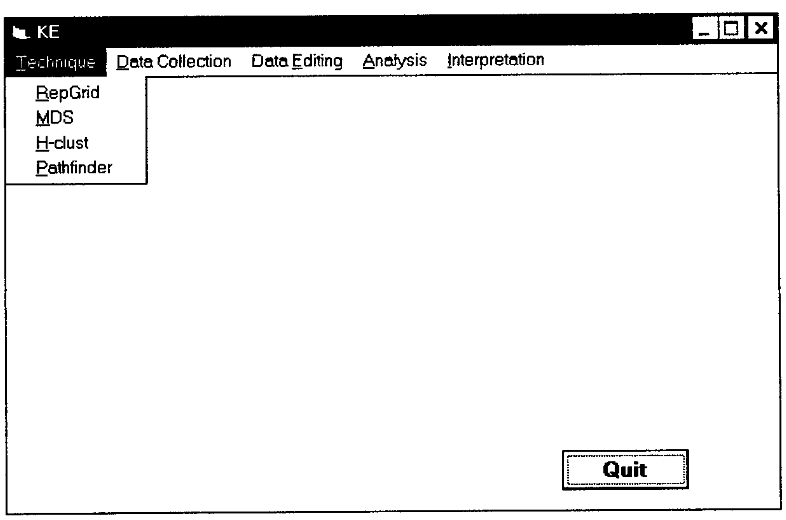


Figure 23. User interface for the selection of indirect technique

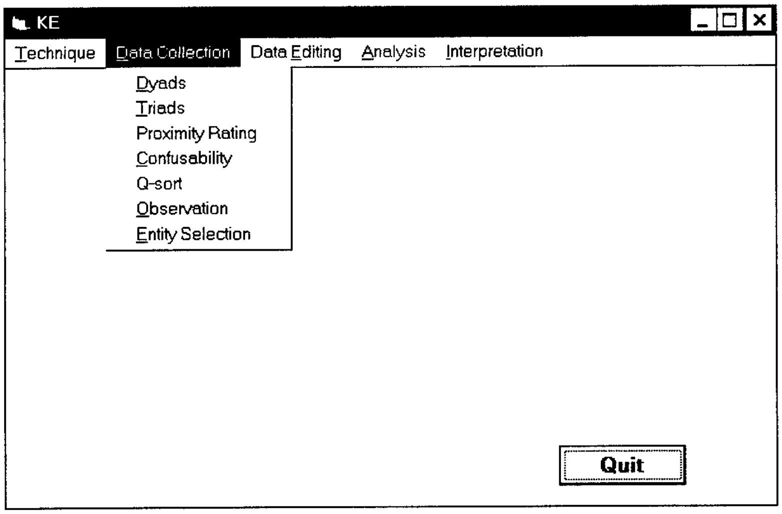


Figure 24. User interface for the selection of a data collection method

3.3.2 Visualization Support for Knowledge Elicitation.

The Visualization Subsystem of the VIEW prototype plays a critical role in data collection for both direct, indirect, and observation techniques. Both direct and indirect techniques require the use of task-specific displays and figures. In *direct techniques*, whether case-specific interviews or protocol analysis, SMEs are presented with a situation to interpret or a decision to make and asked a series of questions about the problem-solving process or asked to describe their inferencing by "thinking aloud". Both of these techniques require the use of domain-specific information. The more realistic the elicitation displays are, and the more the actual working environment of the expert can be re-created, the more realistic and complete the elicited knowledge will be. In *indirect techniques*, a series of stimuli are presented to the SME who is then asked to compare and contrast them by listing similarities or differences, assess their similarity in terms of a pre-defined scale, or rank or sort them into categories. This process is typically done by the knowledge engineer, either verbally or with the names of the stimuli written on cards.

The VIEW prototype design addresses the issue of realistic stimuli by giving the SME and the KE access to the full set of graphical display and display generation functionality of the Visualization Subsystem. Thus rather than asking the SMEs to *imagine* a situation and then describe how they would react, the workstation enables the knowledge engineer to not only present experts with actual displays depicting the situation, from a wide variety of perspectives (e.g., situation template, satellite display, synchronization matrix, doctrinal templates), but also track their use of related displays as they describe their reasoning process and thereby obtain additional information about the type of knowledge that is relevant and the type of inferencing involved. Furthermore, the flexible editing environment allows the knowledge engineer to create and modify the graphical displays to depict the exact requirements of the situations.

The VIEW prototype can thus serve as a simulation environment, allowing the KE to create a variety of different situations and watch the different problem-solving approaches that the SME selects. This capability is particularly useful for case-based elicitation techniques, such as the critical decision or RPD method (Klein, 1989b), where an unusual situation is presented to the

expert who is then asked to describe his or her decisionmaking. The VIEW prototype allows the KE to actually examine important parameters of the situation in a graphical format, thus creating a more realistic setting. Similarly, during protocol analysis, the SME is given a variety of tasks and asked to "think aloud" as s/he solves them. The tasks can be familiar or unfamiliar, typical or atypical, and can have a variety of time or information constraints. Rather than verbally describing or naming each situation, as is most often the case, the VIEW prototype allows the knowledge engineer to present the task using domain-specific displays depicting the conditions (e.g., situation templates representing different courses of action, doctrinal displays depicting different enemy strategies, etc.). Thus, instead of saying to the expert "now imagine you have to perform the same attack but you only have 2 Bradley vehicles instead of 3 Abrams tanks," the KE or the expert can construct a partial representational template showing the reduced weapons. Again, such depictions of the actual situation help trigger a larger percentage of the cognitive and perceptual processes and help elicit more relevant knowledge than simple verbal descriptions.

3.3.3 Technique Library.

The VIEW prototype design supports knowledge elicitation using both direct and indirect techniques. As illustrated in figure 25, *direct techniques* are supported by providing a flexible environment within which a variety of case-specific displays and scenarios can be constructed and presented to the SME, as discussed above. Such scenario displays support the early stages of the elicitation process, whose main task is to begin collecting domain vocabulary and concepts relevant to the decisionmaking and problem-solving tasks. The collected information must then be organized in a format that is accessible for later analysis. Several desirable functionalities thus emerge from these requirements: providing visual and graphical support for the elicitation process, collecting elicited data, structuring the data in a flexible, browsable format, and modification and editing of elicited data. The VIEW prototype design supports these functions by providing the user with facilities for entering free-form text, for performing assisted text analysis (where the SME highlights relevant words in the free-form text which are then inserted into a domain vocabulary or into a more structured format), for constructing domain vocabularies, concept maps, and other structured representations, and for editing and browsing the emerging structures.

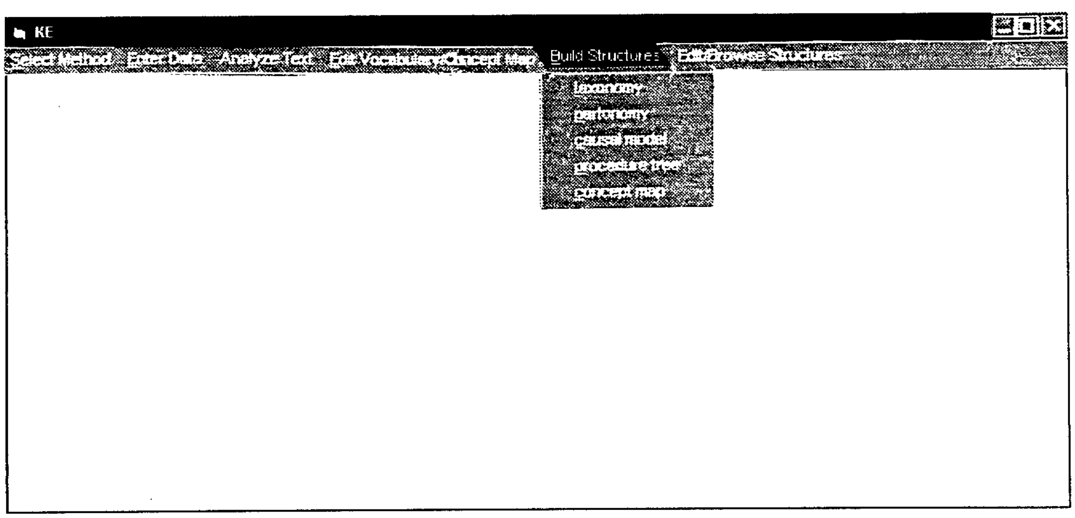


Figure 25. User interface for direct elicitation

The application of *indirect knowledge elicitation techniques*, such as repertory grid analysis or multi-dimensional scaling, is more involved than the use of direct techniques and requires a correspondingly broader range of support from an elicitation workstation. This includes supporting a variety of data collection techniques (both active elicitation and passive observation), providing facilities for editing and combining the elicited data, providing the analytical techniques required to perform the statistical analysis, and providing support for interpreting the results. The design of the VIEW prototype incorporates these functionalities for the most widely used indirect knowledge elicitation techniques: repertory grid analysis, multi-dimensional scaling, and hierarchical clustering.

3.3.4 Data Collection Methods Library.

While the exact method of data collection is a function of the selected technique, all methods share the requirement for displaying domain-specific stimuli to support the elicitation process. The primary support the workstation provides is thus the presentation of a variety of domain entities for specific questions, cognitive probes, similarity/difference comparisons, or direct proximity judgment elicitations. The type of stimuli presented (e.g., different display formats), the exact format in which they are presented (e.g., dyads, triads, sorts), and the type of data collected (e.g., free-form text descriptions, similarities and differences, proximity ratings, transition frequencies during use) are a function of the selected collection method.

3.3.4.1 Data Collection for Direct Techniques.

Direct techniques generally require the collection of free-text or responses to specific questions and probes presented to the expert in dialogue boxes. Different types of interviews are supported by asking the expert a series of questions and recording the answers in dialogue boxes. For example, the critical decision method (Klein, 1989b) is supported by presenting the SME with a series of displays depicting some difficult or unusual situation (e.g., to conduct a mission under particularly difficult weather or logistical constraints) and then asking him or her a series of specific questions ("probes"), designed to elicit information about decision-making under those circumstances. Figure 26 illustrates a sample user interface snapshot for supporting the critical decision method. As the user enters information in the dialogue boxes, the data are stored and can be retrieved for later analysis and editing, either by the KE or the SME. The data can also be structured into abstract representation formats such as various types of semantic nets, production rules, conceptual graphs, etc. It is up to the user to select the desired format. Once selected, the workstation prompts the user to enter the contents of the structure. These can be either entered directly, through dialogue boxes, or can be selected from existing data, that has been elicited through other direct techniques.

The VIEW design provides these functionalities and supports the elicitation process via the following features:

- *Visualization and Graphics Support:* This provides a means of displaying a *variety of scenario and task-specific displays* and allowing the users (SMEs and KEs) to *modify* them to reflect the SME's mental representations and to *navigate* among them in a manner that best reflects the SME'S inferencing process.
- *Text Recording Facilities:* This provides facilities for the entry of free- form text as the SMEs describe their inferencing. The text would be a record of the elicitation process and would serve as data for future investigation of the process. The text would be linked with

the relevant displays, as appropriate, thus creating a hypermedia environment which could later be used for other purposes (e.g., training and tutoring).

- *Assisted Text Analysis Facilities:* This allows the users (SMEs or KEs) to analyze the elicted textual data by supporting an assisted text analysis process, where the user would highlight items of interest in the free-form text and indicate what type of object or structure the text item was describing, and create the corresponding object or augment the evolving knowledge structure.
- *Vocabulary and Concept Map Building Facilities:* A critical component of knowledge elicitation is the construction of a domain vocabulary, consisting of objects and attributes. This function supports this process in two ways: allowing the users to directly enter objects and their descriptions into the database, and allowing the users to highlight words in the elicited text and transforming these into objects automatically.
- *Assisted Structural Analysis Facilities:* Object collections and conceptual maps are useful first steps in the elicitation process, but do not reflect the complexity and variety of internal mental representations, nor do they easily correspond to specific visualization formats. The raw data represented by these collections of entities must be assembled into meaningful structures before they can be said to reflect the internal mental organization of human experts. The assisted structural analysis facilities of VIEW would assist the user in aggregating the domain primitives into meaningful structures, by allowing the users to select objects in the data base and organizing them, using a set of link-types, into higher-level structures, such as taxonomies, partonomies, causal models, procedure hierarchies, decision trees, etc.
- *Dynamic Structure Editor and Browsing Facilities:* Knowledge elicitation is necessarily an iterative process. An ability to edit the evolving knowledge structures is a critical aspect of the process. VIEW supports the users in this by allowing them to browse and modify the emerging knowledge structures.

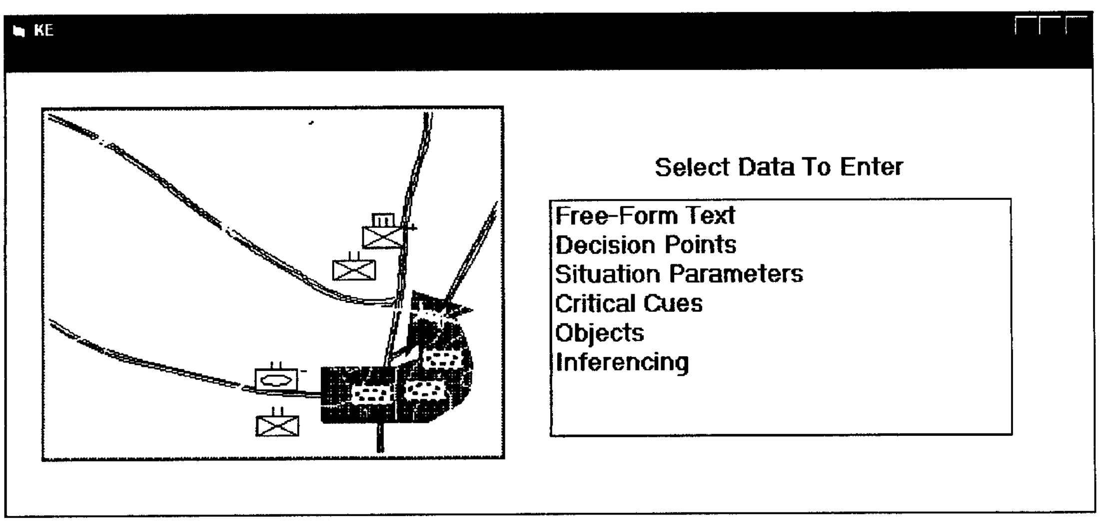


Figure 26. User interface for supporting critical decision method

3.3.4.2 Data Collection for Indirect Techniques.

Indirect techniques require more elaborate data collection methods. Although these techniques vary in the final structures they produce, and the details of the analytical process, they share many of the data collection techniques and can often share intermediate data formats. The KE user interface is therefore structured in a way that emphasizes these similarities. The main menu bar consists of the top-level steps in the application of the indirect techniques, with the pull-down menus providing the choice of action applicable in the current context. Figures 3.3-1 and 3.3-2 shown earlier illustrate this menu structure.

One of the most tedious aspects of the indirect techniques is the preparation of the appropriate entities for comparison and their randomized presentation. The VIEW prototype supports both of these functions via a flexible selection of domain entities and their presentation in a variety of formats for eliciting different forms of data required for analysis. These include direct proximity assessments, dyad and triad comparisons for eliciting similarities and differences, and a variety of observational techniques, where different stimuli are presented to the subject who is then asked to identify some shared property, or to order the stimuli according to some ranking criteria. Entities which can be used for data collection include different COAs, different avenues of mobility within a particular terrain, different doctrinal templates, different situation templates, different weapons placement, etc. In addition to these high-level entities, the workstation can also provide simpler stimuli (e.g., different icons). The range of complexity of the stimuli thus varies widely, thereby providing an additional set of choices for the knowledge engineer and supporting an empirically-based approach to knowledge elicitation, where a variety of stimuli and data collection techniques can be investigated.

The elicitation screen layout differs for the different data collection methods applicable to different indirect elicitation techniques. For example, for repertory grid analysis dyadic entity comparison, figure 27 shows how the screen displays the two entities being compared and prompts the user to enter similarities and differences between them. In multi-dimensional scaling the data collection may involve proximity assessments of two stimuli. Figure 28 shows the screen layout supporting the needed proximity assessment elicitation.

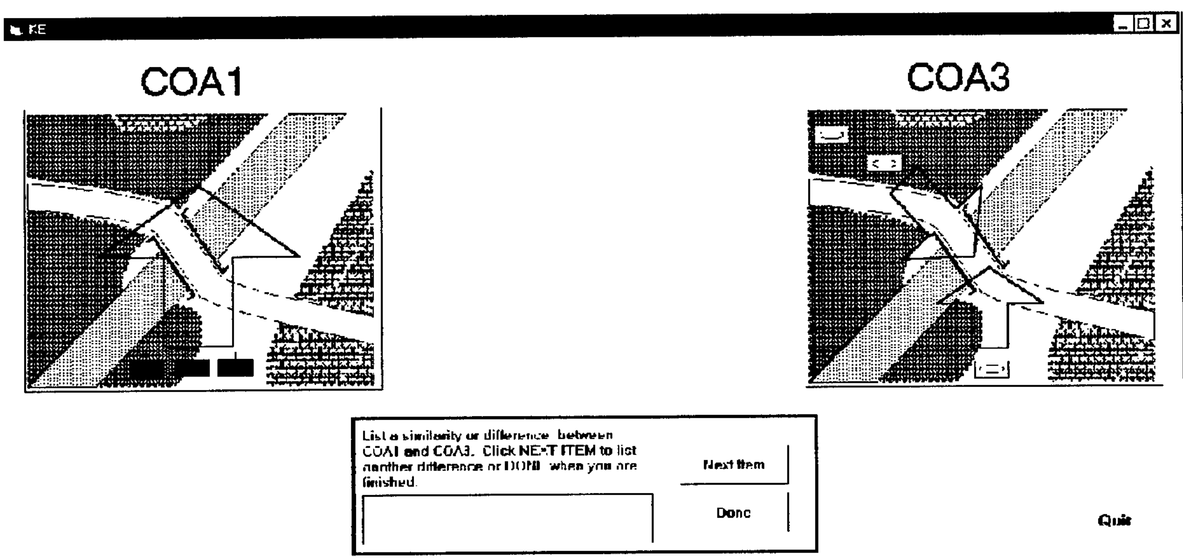


Figure 27. User interface for difference/similarity elicitation for repertory grid analysis

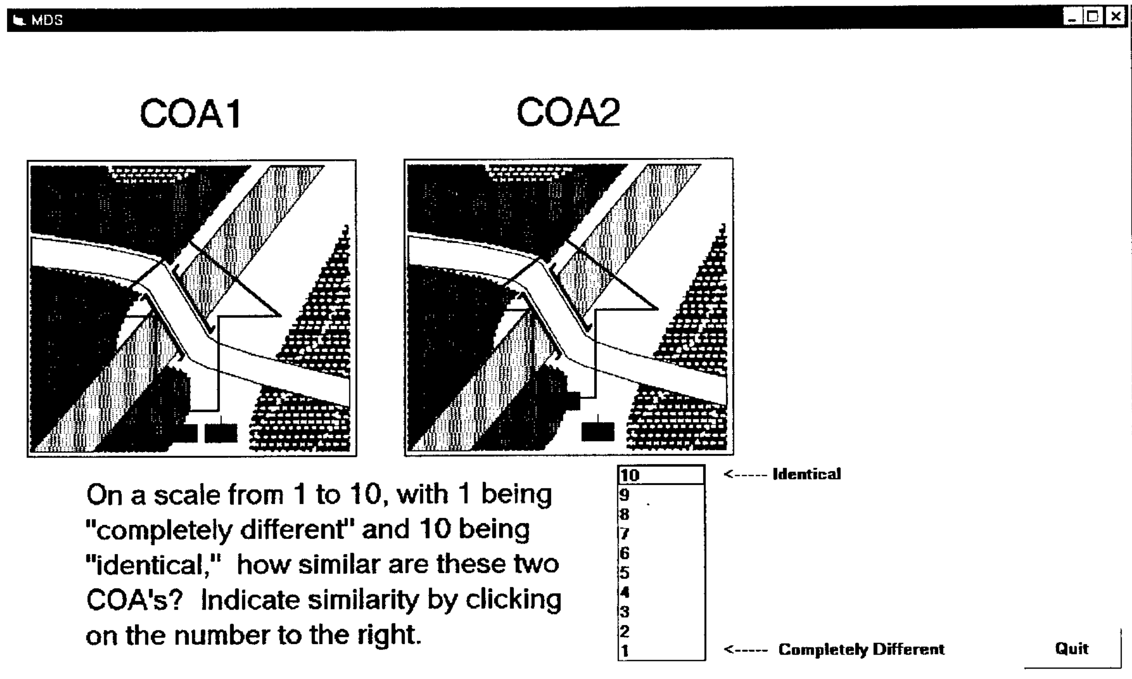


Figure 28. User interface for proximity assessments for MDS

3.3.4.3 Data Collection via Passive Observation.

An important alternative data collection method facilitated by the VIEW prototype is *non-intrusive observation of user behavior*. In contrast to direct elicitation, where SMEs are prompted to provide proximity assessments or various types of sorts and rankings, indirect observation records the SME's behavior over time and infers from these observations the proximity measures necessary for MDS and other indirect KE techniques. The VIEW prototype is an ideal environment in which to gather observation data by tracking the SME's use of the different display formats, templates, and overlays.

For example, in attempting to design new displays that better match the decision-maker's mental models, we may wish to determine which types of information are combined together into single parameters and manipulated by the expert (e.g., speed of movement and current terrain combine into mobility characteristics, etc.). The SME might not always be aware of performing such combinations and yet these combined criteria play a critical role in the SME's situation assessment and decisionmaking activities. By providing a flexible user interface that allows the user to select from a number of possibilities, the workstation can track display use during a number of diverse decision-making activities, and from these data infer the similarity of individual display formats or the complementary of distinct overlays.

3.3.5 Interpretation of Results.

If an indirect technique is selected by the user, further analysis and interpretation is required. Once the data are collected, the analysis can be performed and the results presented to the SME and KE for further interpretation. For example, if MDS is selected, the VIEW prototype can present the user with the solutions for the different dimensions and indicate the stress level achieved in each case, giving the user the opportunity to select a particular solution. In the case of

solutions with more than 2 dimensions, the system would present a series of 2-D projections of the data to facilitate interpretations. At this point the SME can be presented with the MDS solutions and asked to label the axes. Figure 29 shows the screen layout for eliciting such an MDS interpretation.

A key feature of VIEW's support of MDS is the ability to display the actual stimuli *during* the interpretation. One of the drawbacks of existing MDS programs is that the solutions are difficult to understand simply because of the formatting of the data. In some cases the individual stimuli are represented by single letters, requiring complex encodings or external manipulation of the data. The workstation would not only provide the MDS plot with the actual entity names, but would also allow the users to click on the entity and display the actual original stimulus. This presentation greatly facilitates the interpretation process. The workstation can also construct 2-dimensional projections and allow the rotation of the solutions, functions which are required when interpreting data from a single SME. All of these manipulations are normally difficult and tedious and not supported by existing software.

Numerous other possibilities for data collection, data editing, and combination of data sources can be supported by the workstation. For example, the user can view the result of hierarchical clustering and select a subset of data (one branch of the tree) to further analyze using MDS. These types of operations, while conceptually simple, are difficult and tedious in practice, and will be supported by the integrated KE environment provided by the VIEW prototype.

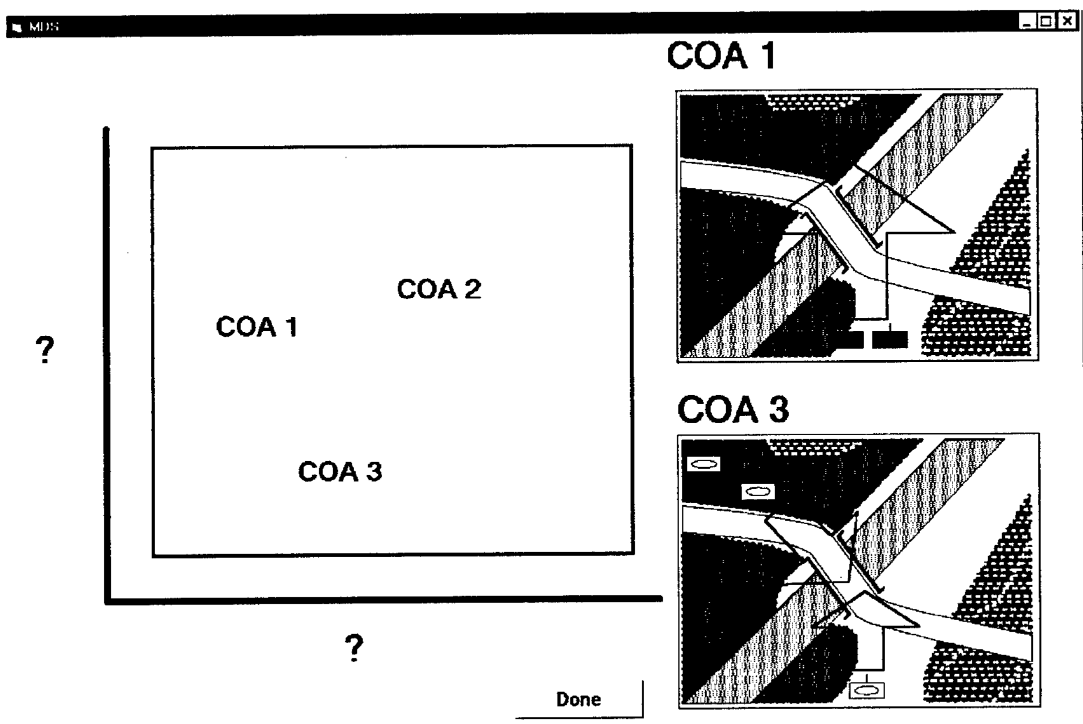


Figure 29. User interface for interpreting the results of an MDS analysis

4. System Operation and Demonstration

This chapter describes operations of the VIEW design, and illustrates capabilities of the prototype system. Section 4.1 provides an overview of the visualization/elicitation process to place the VIEW functions in context. Section 4.2 then describes a sample tactical scenario used to focus the KE effort and the prototype development effort. Section 4.3 proceeds with an example visualization session conducted by the commander during mission planning to illustrate VIEW visualization functionality. Section 4.4 complements this with a description of some of the knowledge elicitation sessions conducted to support the VIEW prototype development effort.

4.1 General Overview of Visualization/Elicitation Process

Figure 4.1-1 illustrates how the visualization/elicitation process is divided into distinct steps: definition of a tactical scenario used as the basis of case-based knowledge elicitation; the knowledge elicitation process itself (both direct and indirect); design of the graphical displays depicting the elicited models; and development of the graphical prototype that allows navigation among the individual displays and supports further mental model elicitation and refinement. The final stage of this process is the interactive manipulation and refinement of mental model visualizations by the battlefield commander.

It is important to keep in mind that while the diagram in figure 30 shows the distinct steps in the process as a linear sequence, the actual visualization/elicitation process is interactive, with many feedback interactions between the various stages, including the final prototype testing stage and the first scenario development stage. In other words, it is reasonable to expect that as a result of visualizing the sequence of models involved in a particular tactical planning situation, the user will further explore the knowledge structures and inferencing supporting some of the decisions made, and to this end will make modifications to the scenario to further explore particular types of decisions. These modifications may then require further knowledge elicitation sessions that will result in new or modified display designs. It is also possible that the results of an *indirect* KE session may require further *direct* KE to elaborate the information and knowledge involved, and that issues encountered in the design of the mental model visualizations may require further knowledge elicitation.

In other words, while the end products of the process are visualizations of the commander's mental models, there is information flow and interaction among the intermediate stages of elicitation, display design, and visualization sessions.

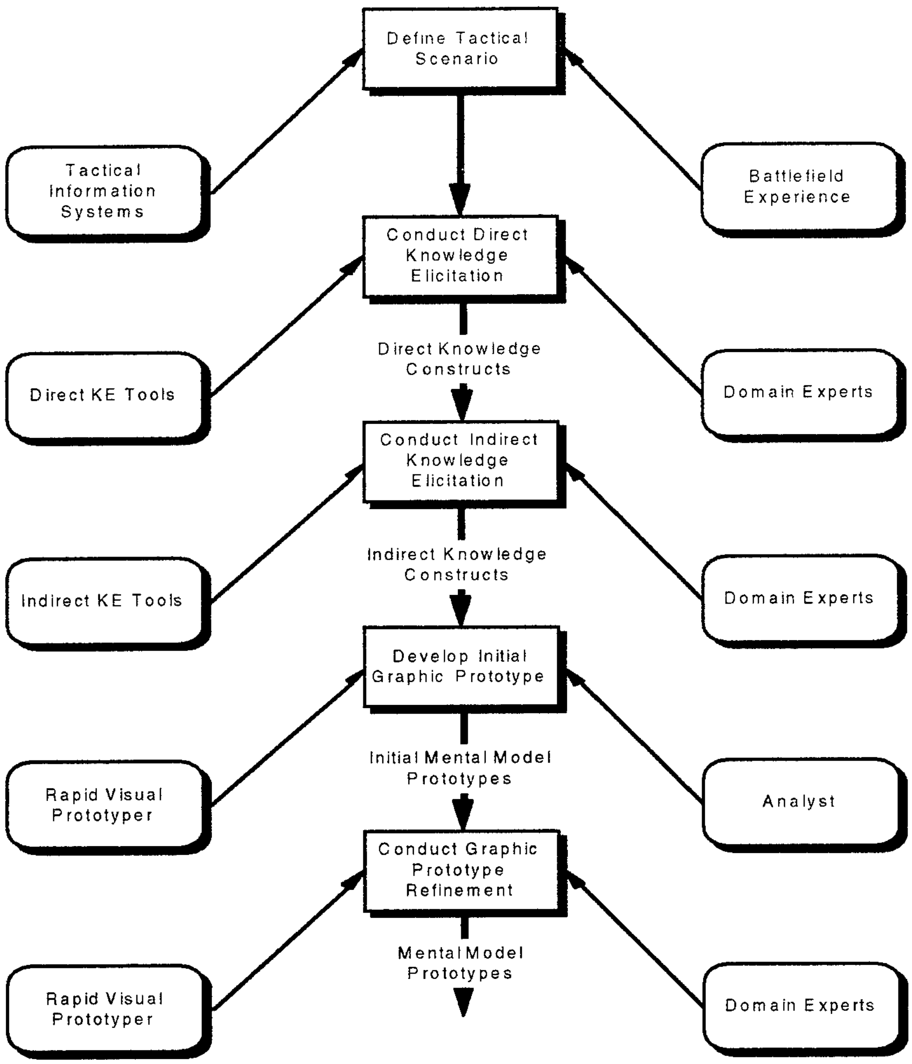


Figure 30. General overview of visualization/elicitation process

4.2 Scenario for Demonstration

To demonstrate VIEW functionality we selected an offensive force-on-force medium intensity conflict scenario using terrain maps from FM 34-130 (The Intelligence Preparation of the Battlefield). The scenario was constructed by subject matter experts, both military intelligence officers, one with a background in Air Defense Artillery (ADA). The scenario and the decisionmaking take place at the brigade and battalion levels.

Figure 31 shows the scenario terrain and force composition, as currently visualized through one window of the VIEW prototype. Although the information density is relatively low compared to conventional map/overlays (due to time constraints of the project) it illustrates the basic functionality provided by one visualization function of the prototype. For the scenario illustrated, the mission, given by the division commander, is to attack and penetrate enemy forces which are assembled on the opposite side of a river, east of the assembly area. The objective is to take the bridge (indicated by "Obj" in figure 31) and proceed to phase line WASP, beyond the bridge to the east. The METT-T format of the mission is shown in table 1.

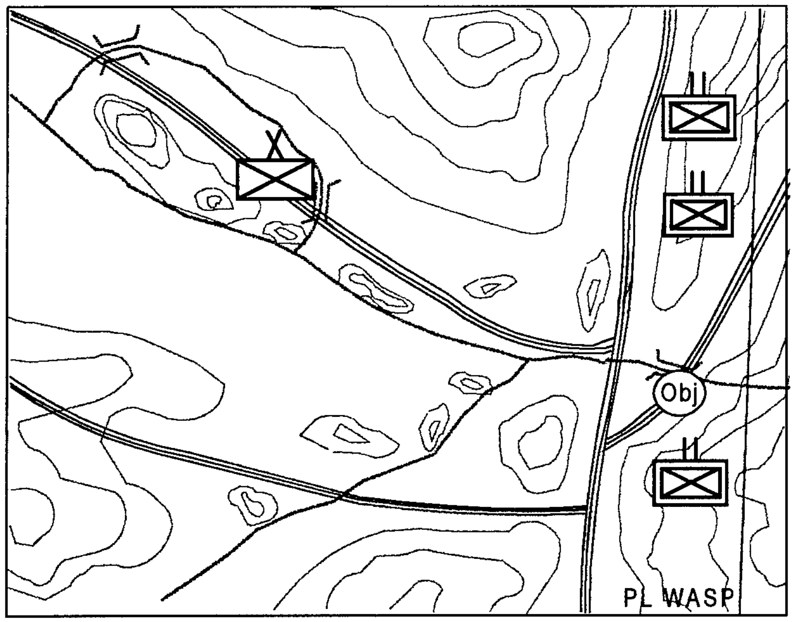


Figure 31. Scenario terrain and force composition

Table 1. METT-T format of the scenario mission

Mission:	Attack and penetrate 1st echelon enemy forces
Enemy:	The enemy troops consist of mechanized infantry battalions in offense
Troops:	Brigade consisting of 3 mechanized infantry battalions and 1 mechanized armor battalion
Terrain:	The terrain is partially forested, sparsely populated, with slight rolling hills, several rivers, major roads, and bridges.
Time:	The brigade is given 48 hours to successfully execute the mission and proceed to phase line WASP.

The global decisionmaking initially takes place at the brigade level. Once the commander is familiar with the terrain, troops, and weapons systems, s/he can begin to plan out possible alternative courses of action (COAs). In this case, doctrinal strategies indicate several possibilities: direct frontal attack (with two alternatives: overwhelming attack and attrition), and a feint attack, involving a two-pronged attack with the objective of seizing the bridge (a feint attack force will engage and hold the enemy in the north, and the main attack force with three armored battalions will move in from towards the bridge from the south).

After examining the available options, the commander begins to explore the feint attack COA by considering in greater detail two alternatives for battalion allocation: splitting the armored battalion or leaving it intact. Following an exploration of these options the brigade commander decides to split up the armored battalion and begins exploring the available options within this alternative, which involve a variety of ruse operations in the north part of the river (e.g., simulated or actual river crossings, sending over of scouts), and various options for the actual attack on the bridge in the south (e.g., keeping armored companies intact, attacking the bridge simultaneously from both directions, etc.)

After considering the situation, the commander selects the following strategy: battalions A and B will go north along the river and feint an attack (with B supporting A's activities), while battalions C and D will go south along the river and constitute the main attacking force, with the objective of taking the bridge (with C supporting D's activities). Figure 32 illustrates this COA, using conventional military symbology.

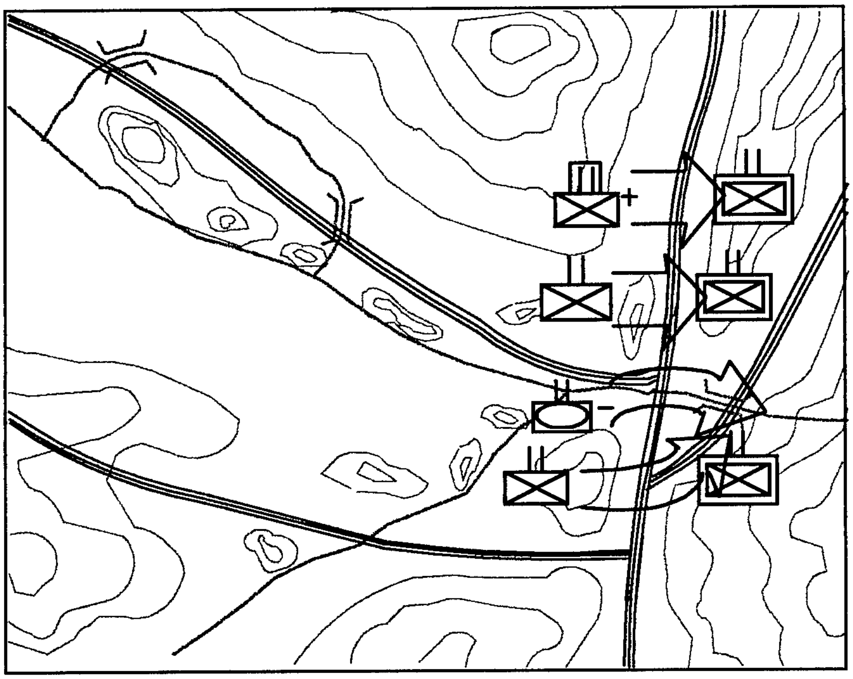


Figure 32. Planned strategy for offensive operations

At this point the planning focus shifts to consider the detailed plan for the two pairs of battalions. Three COAs appear viable for the two battalions whose role will be to conduct a feint attack in the north part of the river: COA1 is to set up several *fake* river crossings, COA2 is to send over several tanks and simulate an actual attack, and COA3 is to send over scouts to simulate preparation for an attack. After considering these alternatives, the commander selects the river crossing COA, where two battalions will be sent north along the river to feint an attack. One will be a task-force battalion, consisting of the three mechanized infantry companies and one tank company from the armored battalion. These forces will engage (fix) and hold enemy defenses by drawing enemy fire and attention to themselves for a specific time period, between midnight-1 am, immediately prior to the main attack.

The commander then decides the detailed actions for the main attacking battalions C and D. These two battalions consist of three tank companies and a mechanized infantry battalion, whose function is to support and reinforce the armored battalion.

Three COAs appear viable for the main attacking force whose immediate objective is to take the bridge. COA1 involves all three companies attacking the bridge from the south. COA2 involves one company attacking from the south while the other two companies follow. COA3 involves one company attacking from the south, one company attacking from the north with the last company following. In all courses of action, the mechanized battalion will relieve the armored companies on the bridge. Once the bridge is secured, the armored companies will move to phase line WASP.

4.3 Use of the VIEW Prototype for Tactical Visualization

To illustrate VIEW functionality, we describe the use of the workstation by a commander in analyzing the above situation and preparing the operations plan to accomplish the stated mission. The script below illustrates a particular path through a large number of possibilities and display manipulations supported by the workstation in order to: 1) demonstrate how the workstation supports the commander's decisionmaking process as he plans and carries out this mission; and 2) illustrate the displays used to support this process. The decision process is described as a series of sequential steps for expository purposes. It is expected that in an actual battle planning situation the commander would shift between different steps and displays throughout the planning process, rather than rigidly follow the fixed sequence shown here. VIEW is designed to support this non-linear visualization and decisionmaking behavior by its hyperlinked set of display options.

The first step in the decisionmaking process for some analysts and commanders is a thorough analysis of the terrain and its impact on both the friendly forces (operations analysis) and the enemy forces (intelligence analysis). This analysis also takes into consideration the current and near-future weather conditions and how they affect the operations.

The map shown in figure 4.2-1 earlier serves as the fundamental display in this step, with a number of overlays providing more detailed views of different aspects of the terrain, at varying levels of detail. Satellite images of the area or depictions of the area from different perspectives may also be used here to augment the symbolic displays. The overlays relevant at this point include: population centers, vegetation types, soil types, friendly troops, enemy troops, supplies (food, ammunition, etc.), friendly and enemy weapons (ranges, mobility). This functionality is discussed in 3.1 and an illustration of the typical interface is shown in Figure 37.

Many of the symbols on these overlays can be viewed at several levels of abstraction with the higher-resolution display shown in a separate window. For example a unit can be *clicked-on* and expanded to its constituent units, a population center can be expanded to view the street maps, and rivers can be expanded to examine current speed, depths, widths, slopes and vegetation on the banks, etc. Figure 33 illustrates how a battalion can be expanded to show its constituent companies, via a direct click on the graphical symbol representing the battalion. Note also how the terrain scale changes to accommodate the invested unit resolution. Figure 34 illustrates a similar capability in viewing population centers. The interface which allows the commander to define unit hierarchy is shown in figure 33 and figure 34.

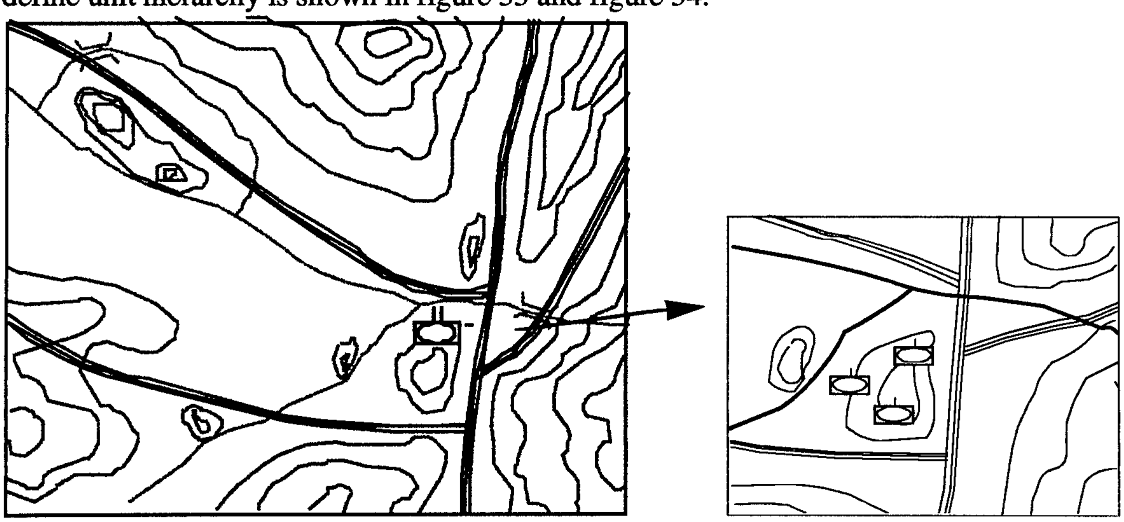


Figure 33. Viewing units at multiple levels of resolution

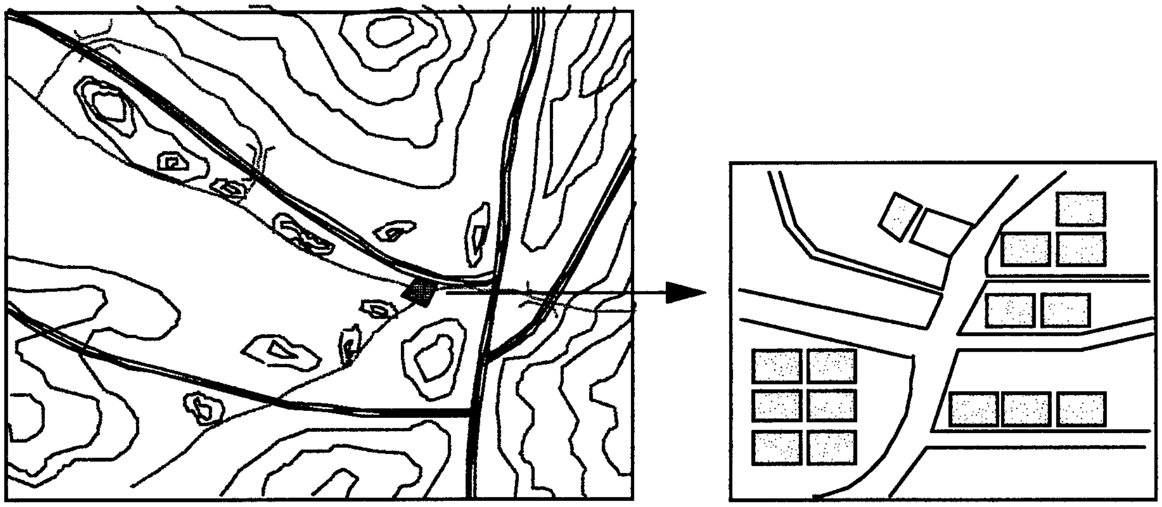


Figure 34. Viewing population centers at multiple levels of resolution

In addition to displaying the standard overlays described above, the user can also query the system to provide customized views of the terrain that indicate *combinations* of factors relevant to the particular context of the current situation. Examples of queries relevant for the current mission are listed in table 2. The first query identifies possible river crossing areas, in which the depth is less than x feet, the slope is less than y degrees, the vegetation is not dense, the river bottom is not muddy, and the river current is less than z kts. The second and third queries identify potential areas of concealment and corridors of mobility, respectively, and are constructed in like fashion. In the VIEW prototype, the queries are formed by navigating to the query function through the Visualization Navigator Interface (figure 4). Using the query function, the commander can click on multiple pull down menus of many objectives and their attributes to form simple or compound Boolean statements. (This is discussed in 3.1-3)

By performing this query over a designated rectangular region on the topo map, the user obtains a direct visual indication of where, geographically, the query is true. Figure 35 shows the result of applying the concealment query in the rectangular region just west of the bridge objective. The blackened pixels indicate very clearly potential areas of concealment. Figure 36 shows, in similar fashion, corridors of mobility, results from an execution of the third query of table 2.

Table 2. Examples of queries relevant to current mission

Indications of possible river crossing areas:
Areas where (Depth < x) AND (Bank slope < y) AND (Bank vegetation =/= dense)
AND (River bottom =/= muddy) AND (Current < z)
Areas of concealment:
Areas where (Vegetation height > x) AND (Vegetation density < y)
AND (Soil =/= mud)
Corridors of mobility:
Areas with (Roadways of width > x) OR (Areas of width > x)
AND (slope < y) AND (soil =/= mud) AND (vegetation density < z)

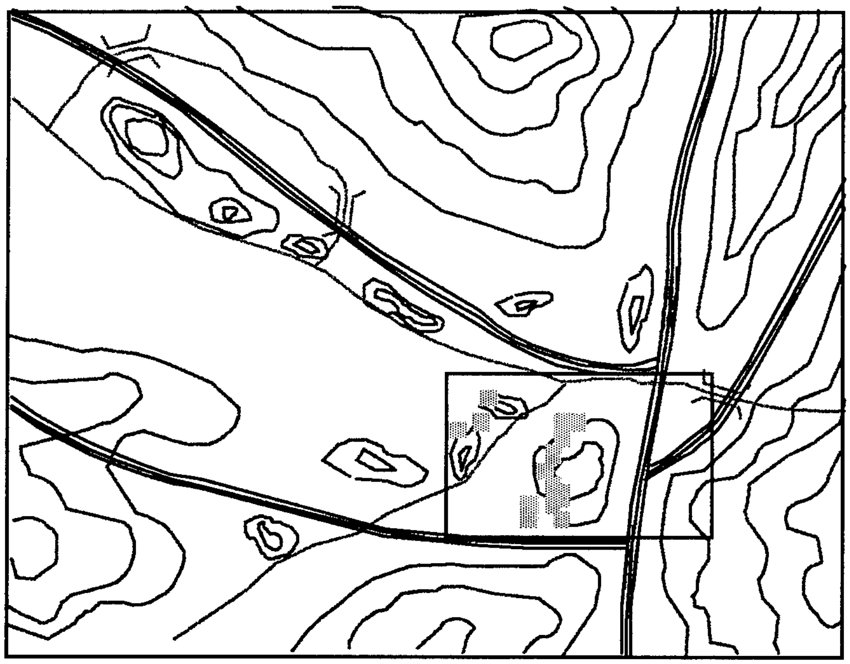


Figure 35. Topo map highlighting areas of concealment

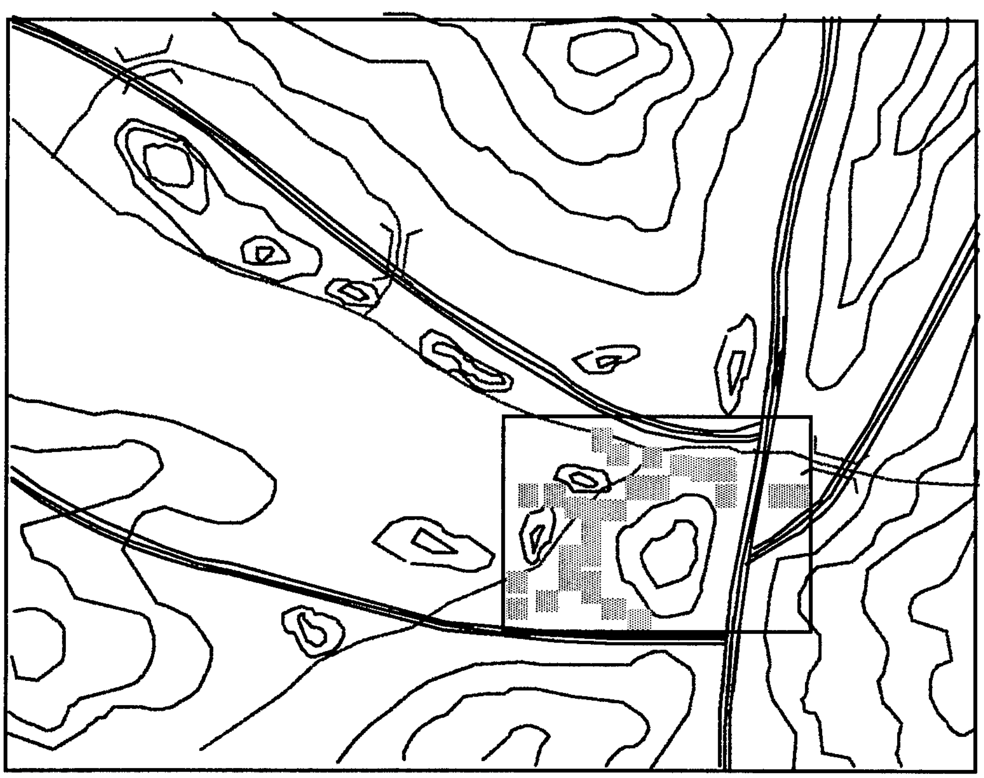


Figure 36. Topo map showing corridors of mobility

In addition to viewing the symbolic representations above, the commander can also view a satellite photo of the same area, by clicking on a selected region. This is illustrated in figure 37.

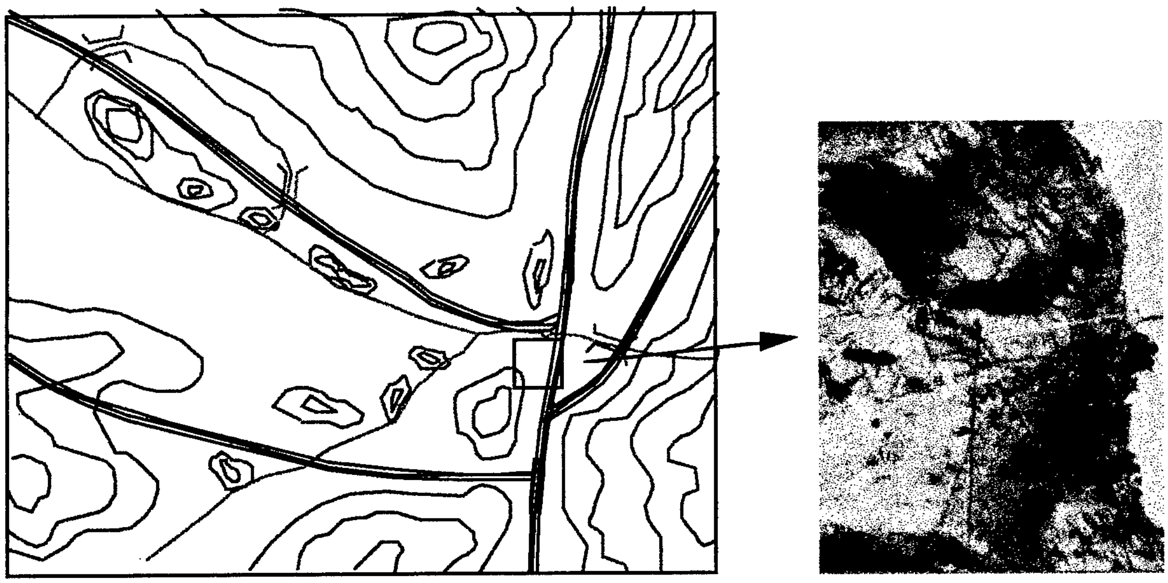


Figure 37. Satellite photograph of selected portion of the battle area

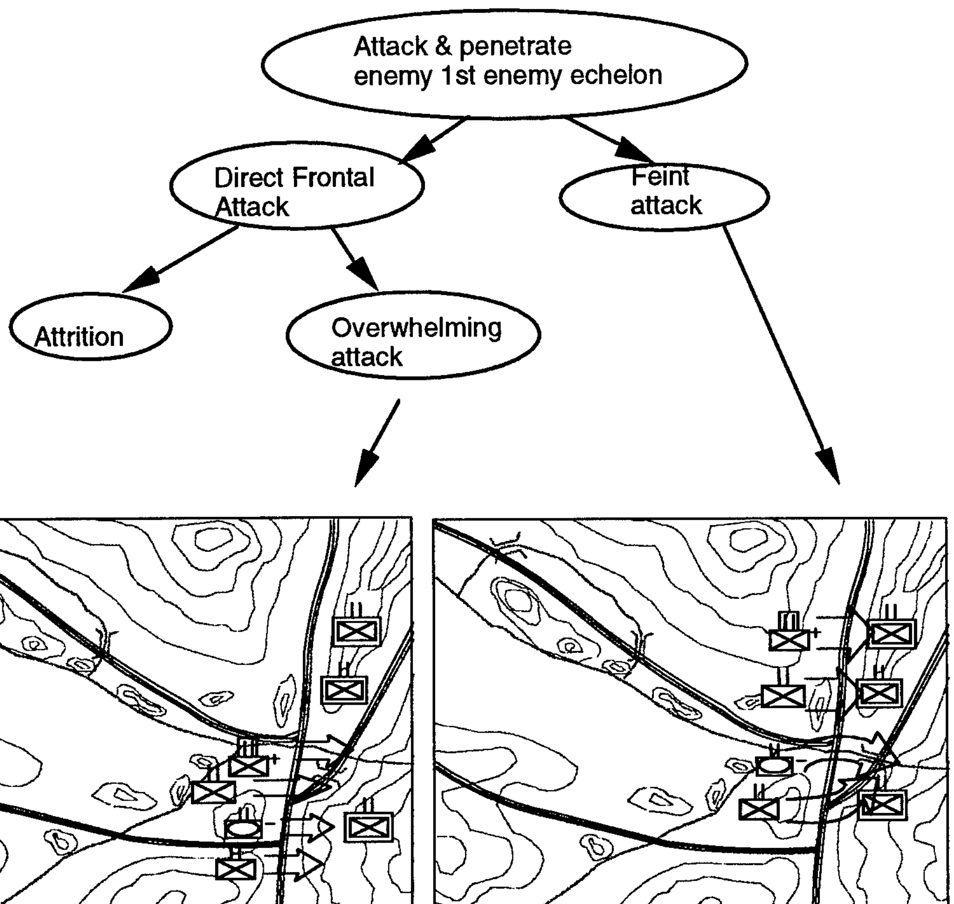
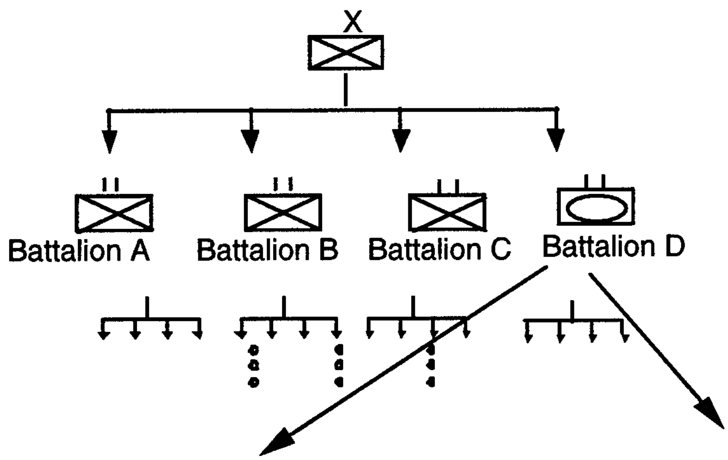


Figure 38: Decision tree and situation templates corresponding to two different decision tree nodes

Once the commander is familiar with the terrain, troops, and weapons systems, he begins to plan out possible alternative courses of action. In this case doctrinal strategies indicate several possibilities (depicted in the decision tree in Figure 38): direct frontal attack (with two alternatives: overwhelming attack and attrition), and a feint attack. Having outlined the high-level possibilities, the commander can view the specific templates corresponding to each possibility. This is done by clicking on the individual decision tree nodes of figure 38 and viewing the corresponding situation template presented in node-specific windows. (The commander can navigate to the decision tree via the interface shown in figure 4).

To make a decision about the form of attack, the commander needs to take into consideration the composition and disposition of both own and enemy troops, the available resources, supplies, and the enemy's doctrinal strategies. A number of displays are available to support this exploration. The commander can click on the unit symbols in the situation displays shown above, to view the detailed unit compositions, for both friendly and enemy forces. Figure 39 illustrates a hierarchical depiction of the units which serves as the base display, to which a number of overlays can be added to indicate a variety of additional information, such as the type of weapons available (via the table in the figure), level of training, percentage of recent replacement, and type of recent activity (e.g., level of rest in last 24 hours/48 hours) (via the bar chart in the figure).



Weapon Systems of Armor Battalion D

TOW	DRAGON	Tank	Mortars 107mm	Mortars 81mm
4	4	4	54	-

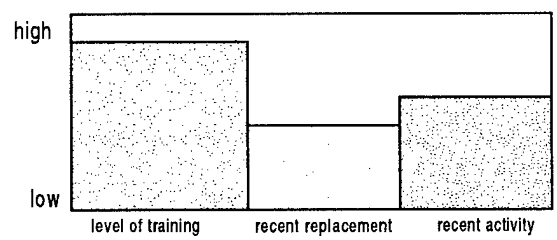
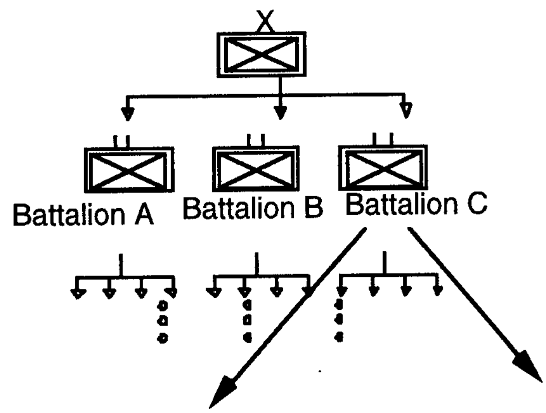


Figure 39. Decomposition of friendly forces & related unit information

Because much of this type of information about the *enemy* forces is uncertain (i.e., has different levels of believability), the commander may wish to see explicitly the different levels of certainty associated with the intelligence information. Figure 40 illustrates a display of enemy forces, indicating the different levels of certainty. The uncertain information regarding enemy weapons can also be displayed by indicating several possibilities (e.g., several types of artillery).

Cross-hatching density is inversely proportional to certainty. The interface showing the top level display panel is shown in figure 9 and the unit hierarchy interface is shown in figure 16.



Weapon Systems of Armor Battalion D

TOW	DRAGON	Tank	Mortars 107mm	Mortars 81mm
3	4 +/- 2	2 +/- 1	31 +/- 6	10

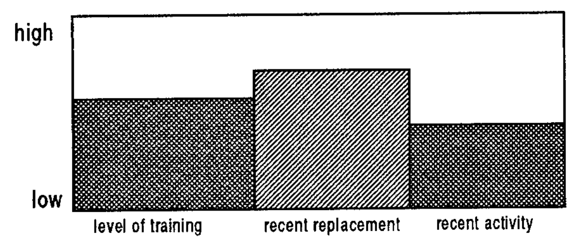


Figure 40. Decomposition of enemy forces, indicating levels of certainty

At this point the commander may view the resources necessary for the different attack options (e.g., ammunition type and amount, fuel, and medical supplies). Figure 41 illustrates the displays showing the ammunition requirements for the two attack options. The display of the projected rate of use of the ammunition supplies clearly indicates the point in time at which minimum ammunition level will be reached, thereby allowing the commander to quickly see how long ammunition will last for the two COAs under consideration.

Based on the exploration above, the commander decides that the feint attack option is the better alternative and begins further elaboration of possibilities within this alternative.

At this point the commander begins to explore the feint attack option by considering in greater detail several alternatives for battalion allocation. The commander begins by adding two more nodes to the decision tree, indicating two alternative battalion allocations: splitting the armored battalion and leaving the armored battalion intact. Figure 42 illustrates this elaboration of the alternatives.

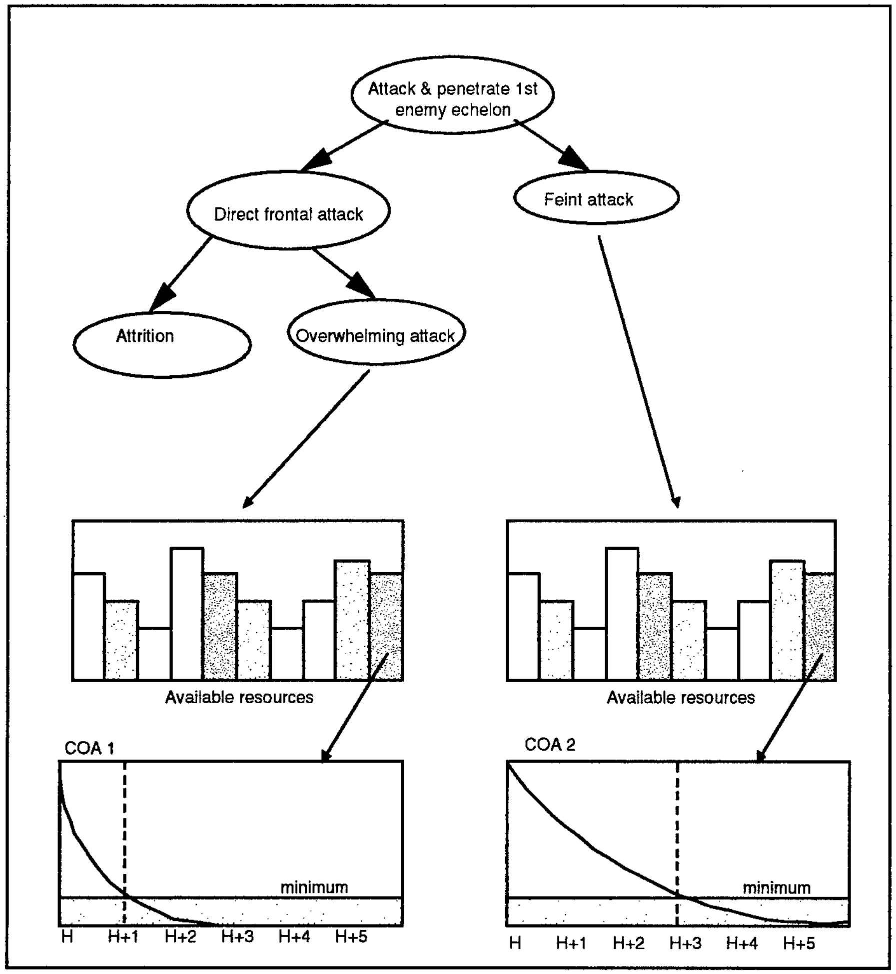


Figure 41. Bar graph of resource use for COA1 and COA2

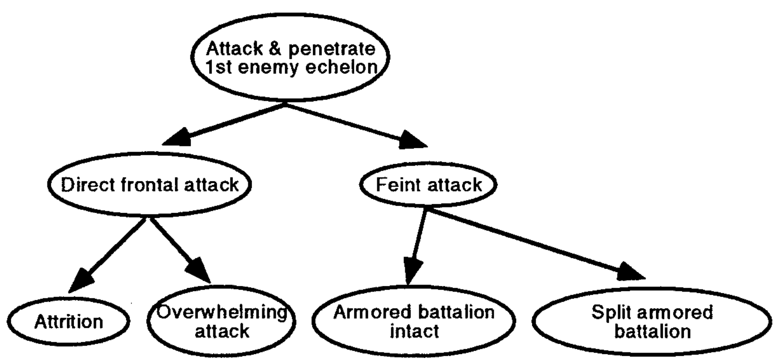


Figure 42. Expanding decision tree nodes to examine particular COA in more detail

The fisheye view displaying the decision tree is intended to capture the commander's evolving representation of the current context. The explored but rejected decisions are still shown (the direct frontal attack and its two variants), as is the path taken to the current decision (the feint attack), but their sizes are smaller relative to the options currently being pursued. This is analogous to a mental representation where the commander maintains a history of most recent activity (that is explored decision options) but de-emphasizes these activities and focuses on the current activity (e.g., exploring the allocation of the armored battalion).

The commander continues to explore in more detail the available alternatives by displaying a variety of information in different formats to decide which of the two allocation options, and the ensuing tactics, is better. S/he may at this point click on the decision-tree nodes and navigate to other alternatives such as splitting the armored battalion for a feint attack.

At this point the brigade-level planning is concluded and the elaborated mission is passed down to the individual battalion commanders as follows: battalions A and B will go north along the river and feint an attack (with B supporting A's activities), while battalions C and D will go south along the river and constitute the main attacking force, with the objective of taking the bridge and proceeding to phase line WASP, with C supporting D's activities.

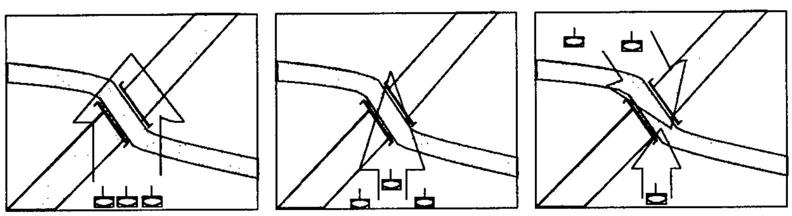


Figure 43. Doctrinal templates for the three COA alternatives

The two battalions consist of a task-force battalion with one mechanized infantry company and three tank companies, and a mechanized infantry battalion whose function is to support and reinforce the armored battalion. Three COA's appear viable for the main attacking force whose immediate objective is to take the bridge, as shown in Fig 43. COA1 involves all three companies attacking the bridge from the south. COA2 involves one company attacking from the south while the other two companies follow. COA3 involves one company attacking from the south, one from the north with the last company following. In all three courses of action, the mechanized infantry battalion will relieve the armored companies on the bridge. Once the bridge is secured, the armored companies will move on to phase line WASP.

The commander can click on each of the alternative COA in the decision tree to explore the situation templates, decision support templates, and event templates to select the best COA. To explore a specific COA in greater detail, the commander selects COA3, as shown in figure 44, to display an expanded view of the area and provide a detailed depiction of the enemy strongholds and positions around the bridge. While exploring COA3 the commander may query the system to identify best crossing points by highlighting points where the current, depth, width, banks, and river bottom lend themselves to river crossing.

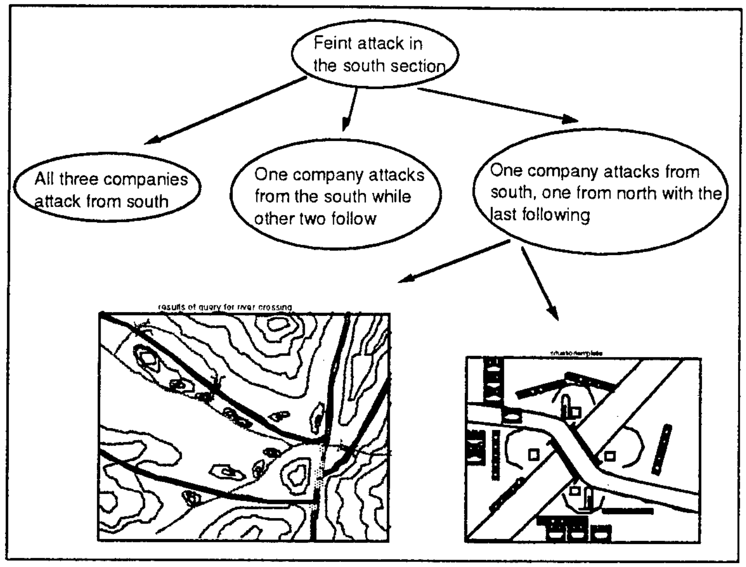


Figure 44. Doctrinal of alternative COAs for battalions C & D

At this point the commander obtains a global view of the operations plan by examining a decision support template, illustrated in figure 45. As shown, this indicates the times at which certain actions will be initiated. By further viewing the corresponding synchronization matrix as shown in Figure 46, the commander can make sure that all activities are coordinated.

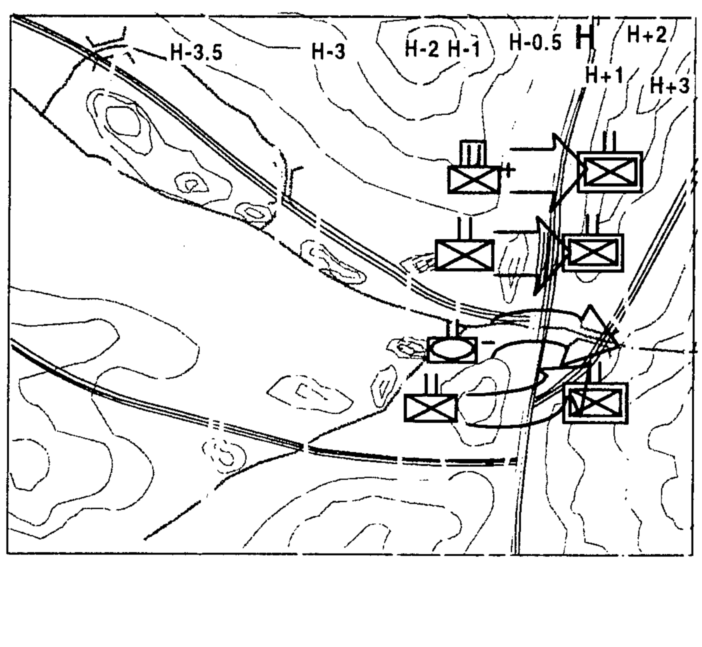


Figure 45. Decision support template for feint attack

	H-3.5	H-3	H-2	H-1	H-0.5	H
Military Police	secure western bridges	control western bridges				
Battalion A			leave assembly area	engineers prepare river crossing		
Battalion B			leave assembly area	engineers prepare river crossing		
Battalion C			leave assembly area		engineers prepare river crossing	
Battalion D			leave assembly area		move to assault area	

Figure 46. Synchronization matrix corresponding to feint attack activities shown in previous DST

4.4 Use of VIEW Prototype for Knowledge Elicitation

The knowledge elicitation companion of the project consisted of three primary activities:

- Conducting both direct and indirect knowledge elicitation with SMEs.
- Extracting from these sessions the requirements for the VIEW prototype.
- Designing the KE subsystem of the overall VIEW prototype.

This section discusses the knowledge elicitation sessions and the functionality of the VIEW design in supporting a variety of knowledge elicitation techniques. The discussion is divided into direct and indirect knowledge elicitation sessions, to illustrate the distinct VIEW functionalities supporting these activities.

4.4.1 Direct Knowledge Elicitation.

A major challenge in knowledge elicitation, particularly direct elicitation, is to find a balance between allowing the SME to speak freely, which enables the knowledge engineer to follow the SMEs reasoning and infer the underlying mental models, and probing for specific information, which allows the KE to identify the distinct knowledge structures or their contents. To address this issue we used several formats of interviews during the KE sessions, including structured interviews, case-based elicitation, inferential analysis, and a modified form of the critical decision method (Klein, 1989b).

In addition to these, we also questioned the SMEs extensively about the nature of their visualizations, about the usefulness of particular displays during decisionmaking and problem-solving, and about desirable modifications to these displays or alternative display designs that would better support their inferencing. This mixed format method of elicitation provided a compromise between free-form unstructured interviews and more constrained, and constraining, forms of structured elicitation.

A total of ten KE sessions were conducted with two SMEs over an eight week period. The elicitation process began with an unstructured interview whose primary purpose was to obtain information about the SMEs background and to establish rapport. The next phase of the process consisted of selecting an elicitation and demonstration scenario. Several possibilities were discussed, including both high-intensity conflicts (force-on-force) and low-intensity conflicts

(operations other than war). A traditional force-on-force scenario was selected for this project, primarily due to the fact that the SMEs experience level was higher in these types of operations. However, representative scenarios from both types of conflicts were used as cases during the early elicitation sessions. The scenarios were selected from the FM 34-130 Intelligence Preparation for the Battlefield, and were modified to fit the force-on-force requirements, where necessary.

The case-based interview format was selected because it is more effective at focusing the discussion, particularly when more than one domain expert is involved, and because it is a more efficient means of obtaining relevant domain knowledge. The elicitation sessions followed two distinct formats: *display-centered* and *decision-centered.*

In the *display-centered* format used in early sessions, the experts were asked to comment on each display depicting the scenario in the field manual. This interviewing format allows us to focus on how the experts used existing visualizations, what information they gathered from each display type, and how their thinking was aided and influenced by the different display types.

In the *decision-centered* format the experts were asked to describe the planning process for the selected scenario and were allowed to use whichever display they found most helpful. They were also allowed to draw their own displays and were encouraged to indicate how the existing display formats were limiting their reasoning and what formats might be more helpful. During the decision-centered interview format a modified form of Klein's critical decision method (Klein, 1989b) was used and the experts were either explicitly probed or their "thinking out loud" was recorded to collect the following information:

- Specific decisions required at different points of the scenario
- *Information and timing requirements* for each decision
- Specific *tasks* that must be performed and *strategies and procedures* available for each (e.g., decide on a course of action and select from the possible types of engagement)
- *Types of inferencing* performed (e.g., combining various sources of info, what-if simulations of dynamic situations, wargaming, constraint checking)
- Information or processes where uncertainty exists and reasoning strategies for *reducing or managing the uncertain information*
- Aspects of decisionmaking and inferencing that were particularly *difficult to perform*

The decision-centered format allowed us to gather a wide range of information about the inferencing processes during battlefield command. By encouraging the experts to draw, use existing displays, or describe ideal displays, we also collected information about points in the reasoning process where displays were helpful and in what ways the existing displays could be augmented to better support human inferencing and reflect actual internal representational structures. This mixed interview format allowed us to collect information about a variety of knowledge structures, inferencing processes, and display formats. The discussion below illustrates how the VIEW design supports this process.

Throughout the session the VIEW design *provides visualization and graphics support* by displaying different aspects of the scenario under consideration, by focusing in on particular situations and showing their graphical representation, and by allowing the expert to draw and modify the displays as required. The VIEW design could support the elicitation sessions by displaying the different COAs for comparison, by allowing the SME draw a number of corridors

of mobility within some terrain and compare these, or by allowing the SME to draw templates depicting specific experiences in the battlefield.

As the SME is describing particular decisions or particular displays, the VIEW design allows him or her to *enter free-form text* in a dialogue box. This text is saved and can be retrieved later for further analysis by either the SME or the knowledge engineer. In our sessions this capability could allow us to enter data as the SME is describing a situation and to organize the text by linking it to different points in the scenario, or different questions and probes. For example, in the decision-based interview the answers to the different questions could be recorded in separate files, which would gradually accumulate the entities or information related to a particular aspect of decisionmaking (e.g., critical cues, decision points, types of inferencing, etc.)

An important step in the representation of mental models is the elicitation of the basic primitives - the entities that comprise the various internal representations. This is analogous to constructing a domain vocabulary in knowledge engineering. The *assisted text analysis* functionality of the VIEW design could provide support in this activity, by allowing the user to highlight items of interest in the free-form text and indicate what type of object or structure the text item refers to. The corresponding object would be created in the VIEW database and would then be available for manipulation by the visualization component.

Once a sufficient set of domain primitives are assembled, they can be aggregated into more complex structures, corresponding to the SMEs internal representations of the domain. The VIEW design supports this through its *assisted structural analysis* functionality. This functionality allows us to construct larger structures from the elicited primitives. For example, the unit hierarchies or the decision trees could be constructed on-line, during the elicitation process, and these formats would then be available for visualization.

Elicitation and representation of internal mental structures is necessarily an iterative process. The *dynamic structure editor and browsing facilities* of the VIEW design supports this iterative process by providing a capability to browse the evolving structures and make edits as necessary, to reflect the newly elicited information.

4.4.2 Indirect Knowledge Elicitation.

To elicit constructs used by the SMEs in tactical decisionmaking, several repertory grid sessions were conducted using different entities. Three entities were selected, at varying levels of abstraction, to capture the SMEs classification dimensions about different aspects of the battlefield situation. At the lowest level, corridors of mobility were compared to elicit dimensions used by the SMEs to represent and reason about the different avenues of approach when making tactical decisions. At the intermediate level, distinct courses of action (COAs) were compared to elicit more encompassing, situation-level attributes and categorizations. Finally, the experts were asked to recall some specific situations they have experienced and asked to think about their decisionmaking process in these situations. The rationale motivating this selection was to use entities which were richly elaborated in the SME's mind, by virtue of the fact that they were personally experienced rather than simply read about, and to access individual decisionmaking criteria.

The use of the different entities yielded correspondingly different results. By far the largest number of interesting attributes was obtained by comparing individual corridors of mobility, as shown in table 3. The comparison of the COAs yielded a number of attributes (see table 4), but most of these were either highly situation specific or had been elicited during the direct interviews.

Much to our surprise, the comparison of the personally-experienced situations did not yield the anticipated results. One SME had a difficult time recalling specific situations and decisions, which was probably due to the SME's background (intelligence) and level of experience (no actual combat experience). The second SME, with extensive combat operations experience, had no difficulties recalling situations, but comparing the different situations did not yield any attributes that were not already generated during the direct interviews.

To illustrate VIEW prototype functionality we now discuss how the design features support the repertory grid elicitation process we just described. The user first selects the KE technique and the corresponding data collection technique, via the menus illustrated in figure 47. Having selected a dyadic comparison, the user is then presented with a menu of available entity types to compare (see Figure 48). The entities can be retrieved from an existing library of displays or they may be constructed using the VIEW prototype graphics capabilities. Having selected the entities, in this case the different COA alternatives for the bridge attack, the user is then presented with pairs of these entities in a randomized order and prompted to enter similarities or differences. This dyadic presentation is illustrated in Figure 49. VIEW systematically presents pairs of instances of the selected entity class until the SME lists all relevant similarities or differences and indicates that the elicitation is complete.

Table 3. Attributes elicited using corridors of mobility as entities

Size
Width
Security of corridor
Length
Capability to fight within corridor
of roads within mobility corridor
Potential speed
Slope of terrain
Terrain consistency
Presence of choke points
Room to deploy before battle objective
Fire power and protection required
Corridor restrictions
Sensitive to weather
Vulnerability
Opportunity for self concealment
Opportunities for enemy concealment
Opportunities for self cover
Opportunity to secure surrounding region
Flexibility of movement
Ability to keep personnel together
Enemy resources required to block approach
Ability to do reconnaissance on corridor

continued

Table 3. Attributes elicited using corridors of mobility as entities, continued

Likelihood of corridor being clear
Corridor is curvy or straight
Amount of coordination among troops required
Safety of surrounding areas
Fire power deployable
Concealment sensitivity to season
Possibility of destroying concealment
Maneuverability in bad weather
Maneuverability in reduced visibility
Safety in reduced visibility
Vulnerability to ambushes
Areas of vulnerability within corridor
Affords discrete approach
Ability to conceal rate of movement
Ability to conceal number of troops
Ability to conceal exact location

Vulnerability to air attack Table 4. Attributes elicited using scenario COAs as entities

Occupy all enemy units
Allows to mass fire power
Ability to employ overwhelming force
Ability to make river crossings safe
Ability of enemy being able to observe operations
Ability to quickly establish presence in enemy territory
Chance of a counterattack
Vulnerability
Ability to react to trouble
Speed
Ability of terrain to mask operations
Range of enemy's field of fire
Chance of securing objective
Ability to synchronize operations

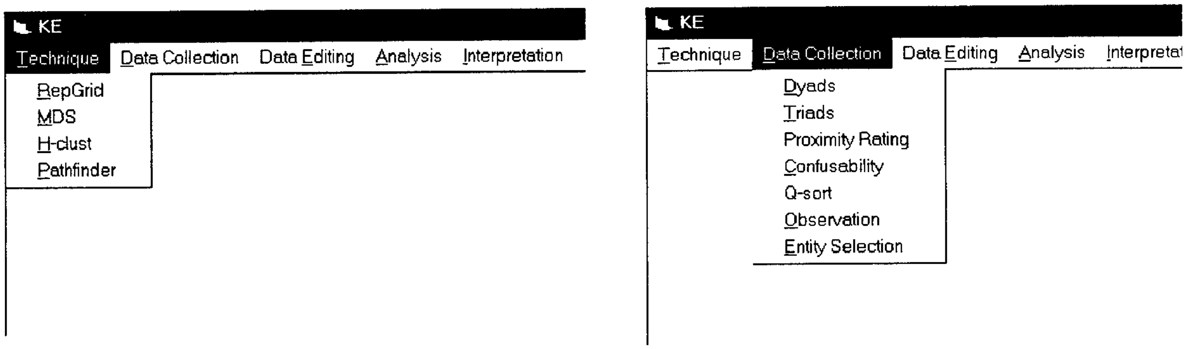


Figure 47. Selection of repertory grid analysis and associated data collection techniques

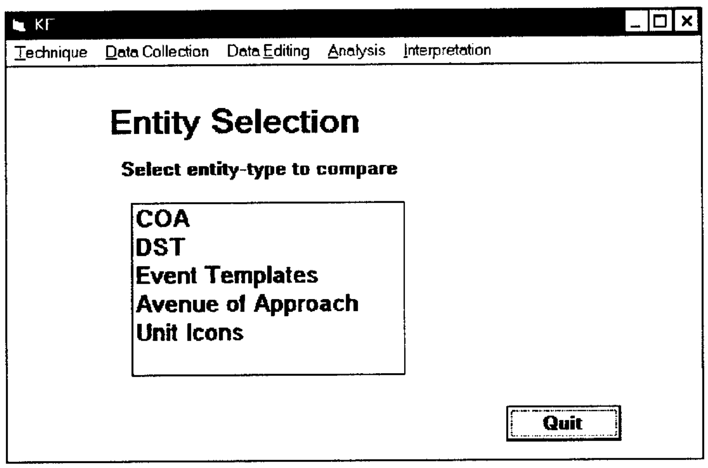


Figure 48. Selecting entities for comparison

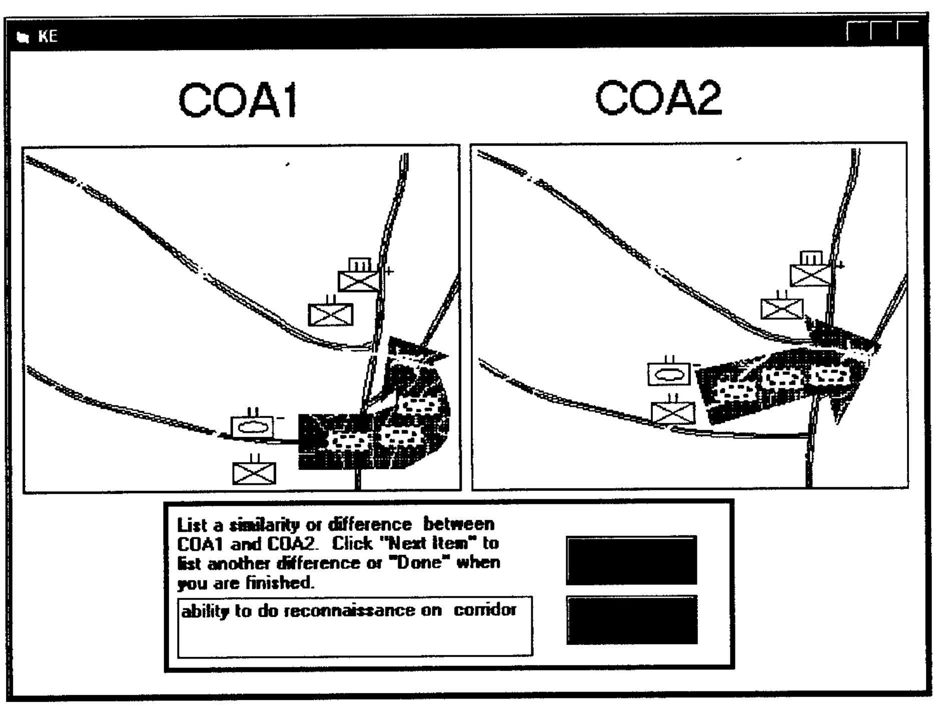


Figure 49. Data elicitation from entity dyads

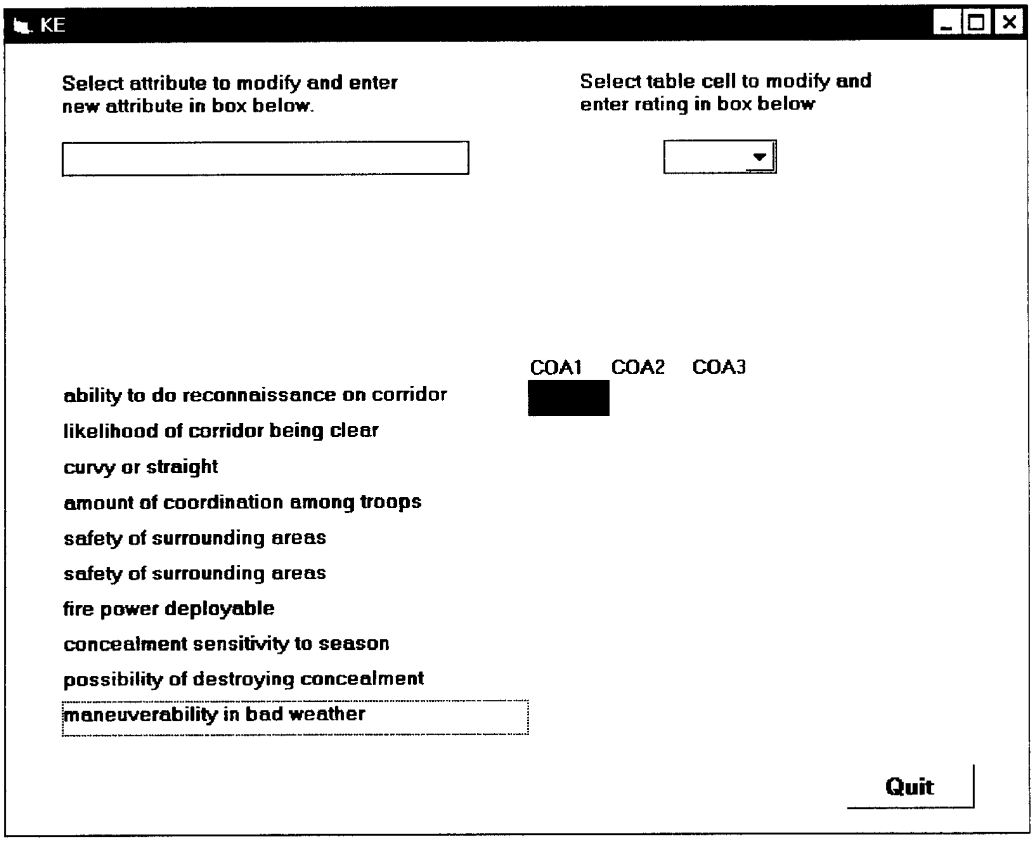


Figure 50. Editing a repertory grid

At this point the VIEW presents the collected data in a table format and gives the user an opportunity to modify attribute names or to add and delete attributes Figure 50 shows the interface for review and editing. By seeing all of the attributes listed at once in this fashion,, the user might realize that *risk* and *vulnerability.* for example, are very similar and might decide to eliminate one of these attributes.

Repertory grid analysis may end at this point, with the elicitation of the SME's constructs (attributes), or it may continue with the elicitation of the ratings of each object along each attribute. If the elicitation continues, then the next step in the process is the collection of these ratings. As illustrated in Figure 51, the VIEW prototype displays an empty grid and prompts the user to enter the ratings. Typically, a 5-point Likert scale rating is used but other scales are possible. In our sessions we used a simple 5-point scale, indicating the meaning of the endpoints (e.g., 1 (low) - 5 (high), etc.). Figure 52 shows a repertory grid with a subset of the actual attributes collected using "corridor mobility" entities.

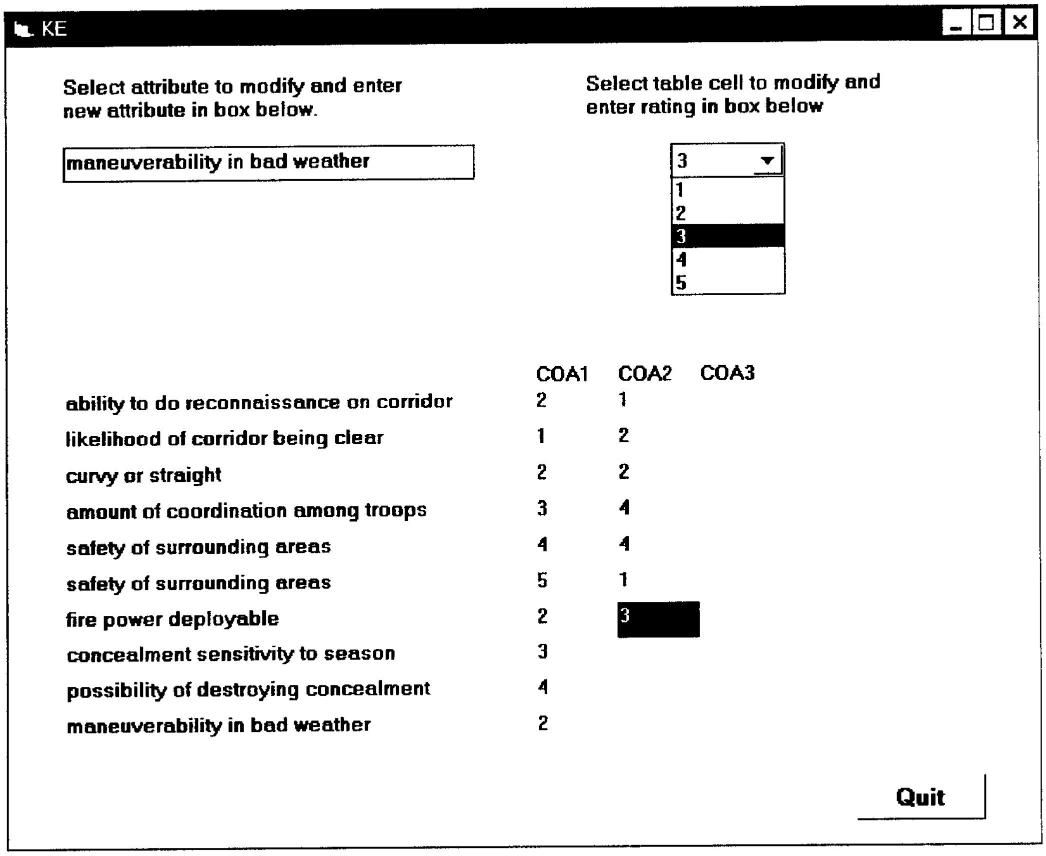


Figure 51. Rating each entity along each attribute

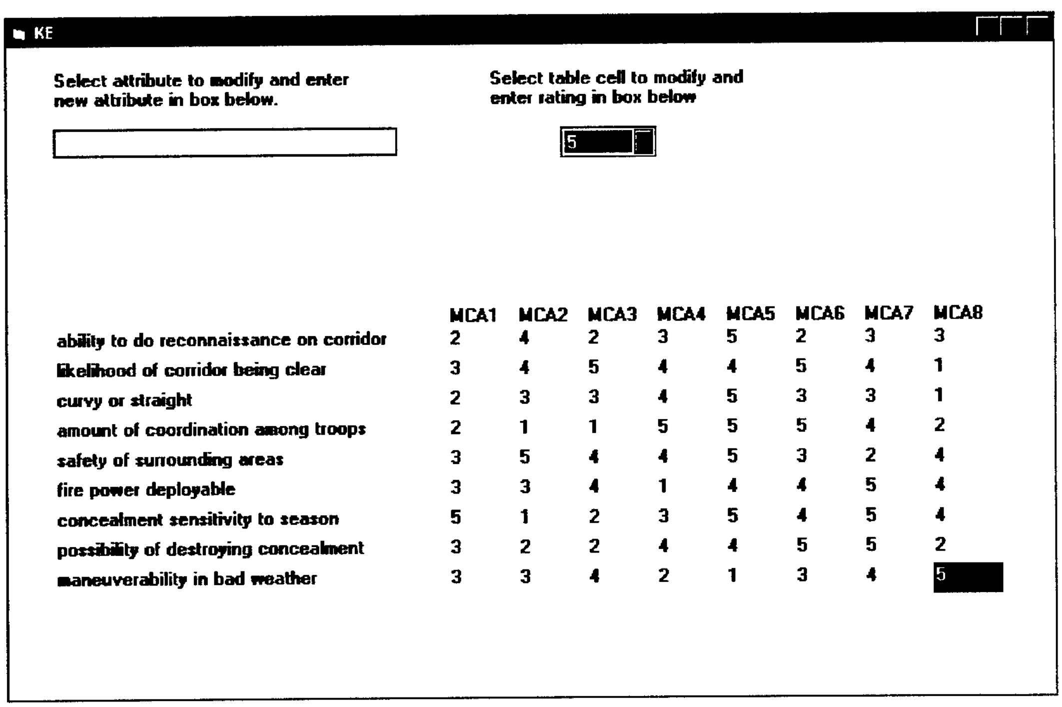


Figure 52. Example of attributes elicited by comparing mobility corridors

At this point the data collection process would be complete and the user could be given an opportunity to further edit the grid, to combine it with other grids (say, from other sessions or users), to apply an analysis technique, or to convert it to a proximity matrix for further analysis. The session then concludes with the user selecting the appropriate analysis technique from the "Data Analysis" menu and examining the results of the process.

4.4.3 Example Sessions Using Indirect Techniques. Elicitation of Different Entities Using Repertory Grid Analysis.

Repertory grid analysis was performed with two domain experts using a variety of entities for comparison. SME #1, an intelligence officer with combat experience, compared three entities. He selected to represent varying levels of abstraction and varying degrees of personal knowledge: three COAs developed as part of the scenario, three personally experienced situations, and eight avenues of mobility in a familiar area of Germany. The VIEW COA displays were used for stimuli, the SME's own drawings of the three situations were used for the three personally-experienced decisions, and a map of Germany with corridors of mobility marked on an acetate overlay was used for the corridors of mobility comparison. The scenario COAs were selected because they represent a relatively high level of abstraction designed to elicit high-level situation-categorizing constructs. The personally experienced situations were selected to capture episodic, idiosyncratic knowledge a commander may have based on individual personally experienced situations. Mobility corridors were selected as an alternative to a) test effect of a fully-detailed map on the elicitation of attributes, b) to compare a larger number of items, and c) to use an entity at a lower-level of abstraction.

The use of these entities yielded the following results: the comparison of the personally experienced situations, much to our surprise, yielded the fewest novel attributes. We expected to obtain idiosyncratic knowledge but instead obtained textbook knowledge and a limited number of attributes. The comparison of COAs yielded 15 attributes, some of which were interesting (e.g., "ability to react to trouble", "ability to amass fire power") and some were not (e.g., speed, vulnerability). The most productive entity was the mobility corridor, which yielded a total of 58 attributes from the two SMEs, with a significant percentage of overlapping attributes. These attributes were interesting in that they a) included non-textbook knowledge, b) included important domain "chunks" (e.g., final approach of avenue of approach), and c) included complex combination of generic and task-specific knowledge (e.g., sensitivity of concealment to weather). Several possible explanations exist for the larger number of attributes elicited using the mobility corridors: 1) a more detailed depiction of the situation (actual map of Germany) was a richer stimulus and thus triggered a larger percentage of information processes, and 2) a larger number of entities was used. Another possible explanation, greater familiarity of the area, could not be valid since both SMEs provided about the same number of attributes but only one of them was familiar with the area.

Characterization of Entity Attributes Using Multidimensional Scaling and Hierarchical Clustering Analysis. The data obtained from the second step of repertory grid process, the ratings of individual entities along each attribute of the mobility corridors, were converted to a proximity matrix using Euclidean distances among entities as determined by square root of absolute difference between each vector of attributes characterizing each entity. This matrix was used as input to both MDS and hierarchical clustering, which yielded the results shown in Figures 53 and 54, providing plots and clusters of the mobility corridors.

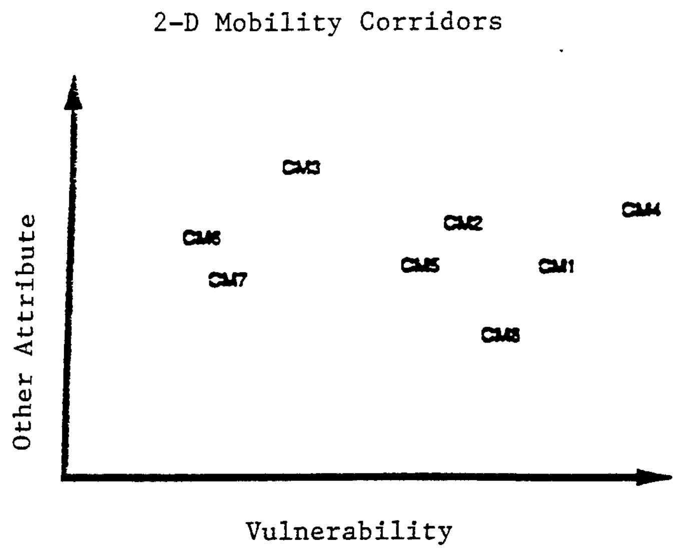


Figure 53. 2-dimensional MDS plot of mobility corridors in a specific situation.

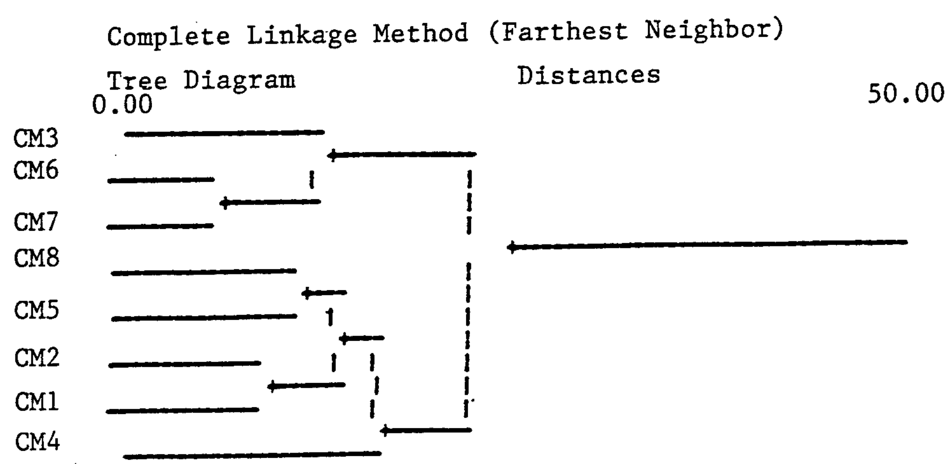


Figure 54. Hierarchical clusters of mobility corridors

These structures were presented to the SME for interpretation, with the hopes of eliciting further attributes that were not elicited previously by the repertory grid elicitation process or by the direct methods. In the case of MDS, this interpretation involved the labeling of axes for 2 and 3-D solutions. In the case of hierarchical clustering, this interpretation involved the labeling of the nested clusters.

The results of the MDS analysis were disappointing in that a) the SME was unable to identify the axes except for one axis in the 2-D solution (identified as vulnerability), and 2) the dimensions elicited had already been identified previously and in any case were not surprising. Hierarchical cluster analysis, while yielding more attributes, was equally disappointing in that none of the attributes were new; the major attribute was "quality of final approach" which had already been identified via the repertory grid elicitation process.

While both MDS and clustering analysis are useful for eliciting structures of entities, they do not seem to be more effective in eliciting classification dimensions and characteristics in the form of entity attributes. For simple attribute elicitation, the initial phase of the repertory grid process appears to be the best method.

4.4.4 Summary of Knowledge Elicitation Results.

The direct knowledge elicitation techniques, case-based display-centered interviews and decision-centered interviews, all provided the data for defining the critical elements of the visualization architecture: object definitions (e.g., terrain, terrain types, environmental objects, map overlays, military templates for depicting situations and decision-making, etc.), display definitions (e.g., maps and overlays, synchronization matrices, decision trees, bar graphs, process diagrams, etc.), and query and rule definitions (e.g., definitions of specific constraints representing high-level cognitive and perceptual constructs of interest to the commander). In addition to these data, the display-centered techniques provided information about the desired types of displays and their use during battlefield visualization. Examples of desired display types and functionalities included the following: ability to view a 3-D terrain representation from arbitrary perspectives, ability to combine and display a variety of weapons and electronic equipment characteristics, ability to support wargaming and what-if simulations through animation, automatic overlay and comparison of event and situation templates to quickly detect differences between predicted and actual situations, and the ability to zoom within an area and rapidly move among different levels of abstraction. Due to the limited scope of this initial effort many of the suggested display formats and display manipulations could not be implemented. However, the information generated using the display-centered elicitation method is included in the recommendations for the follow-on Phase II effort.

The indirect knowledge elicitation effort, which focused on repertory grid analysis, yielded a number of classification attributes relevant for battlefield visualization. These attributes were elicited using different courses of action and different corridors of mobility. Examples of elicited classification attributes are: fire power deployable, concealment sensitivity to season, possibility of destroying concealment, maneuverability in bad weather, maneuverability in reduced visibility, safety in reduced visibility, vulnerability to ambushes, areas of vulnerability within corridor, ability to conceal rate of movement, ability to conceal number of troops, and ability to conceal exact location.

While some of the attributes were also obtained through direct elicitation, the repertory grid method generated a large number of complex constructs quickly and easily. We therefore recommend it as an effective and efficient means of obtaining complex cognitive and perceptual constructs. Our experience with using just two entity types for comparison and generating over 60

attributes, many of which represent complex tactical constructs, indicates that repertory grid analysis is a powerful technique for eliciting the commander's mental model attributes and warrants further exploration. A major feature of the elicitation component of the VIEW prototype design is a flexible means of presenting graphical entities for comparison during the initial stages of the repertory grid process. The VIEW prototype thus promises to be a powerful tool for eliciting a wide variety of tactical constructs, which can then be translated into visual format using the visualization component of the VIEW prototype.

5. Summary, Conclusions, & Recommendations

This chapter summarizes the key tasks conducted under this effort, presents the major conclusions, and outlines the recommendations for a Phase II development effort.

5.1 Summary

The approach taken under this effort focuses on developing a concept design and demonstration prototype which integrates model elicitation and visualization, and which is specialized for the domain of the battlefield commander. Six specific tasks comprise our effort:

- Definition of Scope of Demonstration,
- Review of Knowledge Elicitation Techniques and Software,
- Review of Rapid Prototyping Visualization Software,
- Development of VIEW Concept Prototype,
- Demonstration of VIEW Concept Prototype,
- Requirements Specification for Military/Commercial Development.

We first defined the scope of the demonstration for this feasibility evaluation by reviewing an extensive collection of military material such as Army Field Manuals and several documents from the Defense Technical Information Center. The subject matter of the material ranged from military intelligence and operations to mental models of commanders. By consulting our subject matter experts (SMEs) on numerous occasions, we developed a candidate scenario on which to focus our demonstration. Several scenarios were considered such as operations other than war (low-intensity conflicts) and force-on-force offensive operations (high-intensity conflict). We selected the force-on-force scenario because it provided an adequately constrained but sufficiently rich domain in which to demonstrate the functionality of the VIEW prototype. By conducting several follow-on knowledge elicitation sessions with our SMEs, we were then able to fine-tune our prototype to support key knowledge elicitation functions.

We then reviewed knowledge elicitation techniques and tools and evaluated candidate techniques for implementation. A literature search was conducted specifically focusing on KE techniques that could directly support the specification and visualization of the commander's mental model of the battlefield. Both direct and indirect techniques were reviewed, and evaluated in terms of their ability to identify key components of the commander's model, their reliability, and their ease of use. In addition, we reviewed the availability and capability of associated software tools, to assess their potential for inclusion in a KE *toolkit*, to support computer-based elicitation sessions.

We then reviewed rapid prototyping visualization software options, for potential incorporation into the prototype. Based on a review of the visualization requirements called for in the demonstration, and a review of the KE requirements for commander mental model elicitation, we evaluated potential options for visualization software. The objective was to focus on packages

which could be used for rapidly prototyping and displaying graphical objects, in an object-oriented environment that assures full connectivity between objects and their specific graphical visualizations.

We then developed a prototype visualization/elicitation tool, to support a demonstration of its use in the selected scenario. The prototype included specifications for interfaces to the KE tools selected for elicitation, as well as example visualization displays/controls implemented via the selected visualization software. An overall architecture was developed to integrate both the KE tools and the visualization software, and included a fully relational object-oriented data base to represent relevant objects and object sets comprising the demonstration battlefield scenario.

We then demonstrated the prototype visualization/elicitation tool, to support an evaluation of system feasibility and potential utility in mental model formalization. Primary emphasis was in evaluation of the VIEW prototype's capabilities for visualizing different but related aspects of the tactical scenario, at several different levels of organization and unit resolution. Effort was also devoted to evaluating the VIEW concept design in terms of its ability to support the interactive knowledge elicitation functions needed for mental model inferencing and representation. Functions not implemented in the Phase I VIEW prototype were identified and called out for follow-on Phase II development.

Finally, we specified requirements for military/commercial development of a full-scope tool and methodology. For the military side, we focused on identifying further development and demonstration requirements to be met for a full-scope visualization/elicitation environment for commander mental model representation. For the commercial side, we identified promising commercial market areas, and particular market segments that could benefit from the development of a suitably specialized tool.

5.2 Conclusions

The primary result of this study is a concept demonstration of a visualization/elicitation prototype for graphically representing battlefield commander's mental models.

The major study findings supporting this demonstration can be summarized in the following paragraphs:

A force-on-force offensive scenario was developed at three levels: brigade, battalion, and company. Our friendly brigade included assets such as mechanized armor and infantry while the opposing brigade included mechanized armor. Included among the tools for scenario analysis were Decision Support Templates, Situation Templates, and Decision Trees. Courses of Action were examined for the friendly brigade and constituent battalions and led up to a high-intensity conflict with the enemy on a section of topography that involved river-crossings and the capture of a bridge.

We reviewed a variety of KE techniques, both direct and indirect. Both types of techniques are applicable to the mental model representation and visualization problem. However, no single technique or technique-type is adequate to capture the full scope of the internal representations. It is therefore necessary to use a repertoire of techniques in-concert. In general, *case-based techniques* are preferred, because they quickly focus the discussion and generate concrete results (e.g., specific objects, specific decisions). Direct structured interviews are effective in eliciting a broad scope of knowledge but may not go deep enough to capture specific inferencing types or specific structures. The simple structured interview is thus best used in conjunction with a more specialized interviewing technique. Two techniques were found particularly well-suited for eliciting commander's internal representations: a modification of Klein's *critical decision method* (1989b) which focuses on factors influencing a specific decision, and a

display-centered method we developed during the course of this study, which focuses the interview process on both existing and desirable display formats.

The major disadvantage of the direct techniques is their limited capability to access knowledge which is not easily articulated by the expert in response to direct questioning. Indirect techniques do not require the expert to be able to directly access their knowledge, and thus represent an important complementary approach to elicitation which focuses on the more intuitive, idiosyncratic aspects of expertise. The two types of techniques are best used in conjunction: the direct techniques mapping out the broad scope of the knowledge structures and the indirect techniques allowing further focusing on specific constructs and substructures.

The review of visualization software for implementing the VIEW prototype focused on three operating systems: Unix/X-Windows, Macintosh OS, and DOS/Windows. Although exceptionally good graphics capabilities are supported by Unix machines, such as the Silicon Graphics Inc. Iris series, the relatively high price/performance ratios eliminated them from further consideration as potential hosts in what could eventually grow to be a large network of low-cost hosts. We thus favored the Macintosh OS and DOS/Windows environments. Although the former provides superior graphics tools, we selected the latter because of the much larger installed base at ARI, and the greater likelihood of integration/networking with existing Army systems.

With the focus on DOS/Windows-based software, we quickly identified four key software packages: Visual Basic for the interface, Microsoft Access for the database, Visio Technical for graphical support, and CLIPS for ruleset implementation. These applications all support Dynamic Data Exchange (DDE), so that the applications can be easily linked together. Since Windows is a multitasking system, many event-driven programs or applications are permitted to run concurrently. The DDE feature of Windows allows an application to directly and continuously exchange data with other Window-based applications that support DDE. Visual Basic is an object-oriented, Window-based programming language that facilitates the use of objects to initiate the execution of different programs and applications. Visual Basic uses the Microsoft Access database engine for its local data update and retrieval functionality. Visio Technical is a software package designed to run with Microsoft applications, and can be used to support development of the graphical interface. CLIPS is software developed at NASA's Johnson Space Center, and can be used for implementing any formal ruleset. It provides a rule/object-based environment in which to develop an expert system.

The VIEW system architecture is defined by two major subsystems: the Visualization Subsystem, and the Knowledge Elicitation (KE) Subsystem. The Visualization Subsystem is composed of three interlinked modules: the Tactical Visualization Interface, the Object Database, and the Object World Model. The Knowledge Elicitation Subsystem is composed of two modules: the KE Interface, and the KE Recording/Analysis Module.

The Tactical Visualization Interface supports the commander in two basic ways. First, it provides him with situation-relevant tactical information. Second, it provides him with the means of directly manipulating the object database, to create or modify the tactical situation. A graphical user interface supports navigation across a range of displays maintained in a display library.

The Object Database provides a common object representation for all visualization/elicitation component of the system, and is directly linked to the Tactical Visualization Interface via tactical commands generated by the user and object attributes sent to the displays. Three general classes of objects are maintained in the object class library: 1) terrain-related objects (terrain elevations, vegetation, roads, etc.; 2) military unit objects (echelons, types, weapons systems, etc.); and 3) ground environment objects (battlefield AO/AI, avenues of advance/approach, etc.).

The Object World Model supports an object-oriented simulation of both friendly and enemy forces operating over a specified battlefield reflecting weather and other environmental conditions. Linked to the Object Database via object commands and states, the module provides a direct means of dynamically modifying the database over time. An object behavior library supports the simulations of friendly/enemy mobility, and, via extension, wargaming capabilities.
The KE Interface supports the knowledge engineer in three ways. First, it provides a means of navigating among the KE techniques, via the control interface. Second, it supports the collection of elicited data from the commander who is interacting with the Visualization Subsystem. Finally, it provides on-line access to the results of KE analysis, to support interactive navigation amongst the displays, as a function of the results of the analysis. A graphical user interface supports navigation across a range of techniques maintained in a KE library.

The KE Recording/Analysis Module implements the actual recording and analysis of the elicited data via direct links to the KE Interface. In addition, to insure close linkage with the Visualization Subsystem, the recording modules also accepts as inputs the Visualization configurations selected by the commander via the Tactical Visualization Interface, as well as "snapshots" of the tactical situation as maintained by the Object World Model.

The VIEW prototype was implemented as a Visio Technical extension. The Visio extension approach to software development involved three interrelated steps. The first step involved creating a specific multi-window Visio workspace by modifying the Visio development environment to the specific requirements of this application. The term workspace here refers to a collection of interactive interfaces that are integrated based on a specific design and hierarchy. The second step consisted of adding functionality to the software and its host environment (i.e. the workspace) by embedding stand-alone and functionally independent executables in the environment itself. The stand-alone executables were developed in the Visual Basic development environment. This Windows-based package is very suitable for fast implementation of software designs that involve multiple interrelated interfaces. Furthermore, this development language has provisions for fast and easy access to databases created in the Microsoft Access application. In addition to linking all objects in the workspace to the Microsoft Access databases, using Visual Basic for developing the executables also rendered the overall environment more flexible for the user. The third and final step in the development process involved adding functionality to various objects in the workspace by building stand-alone Visual Basic executables. These executables perform several types of tasks depending on the nature of the object they are linked to. For example, give the user access to different interfaces, as well as object attributes that would be otherwise hidden from the user. Through these stand-alone codes, object databases are updated whenever the user modifies an object attribute through any of the interactive interfaces.

Three aspects of the VIEW prototype are critical for its usefulness in mental model elicitation and visualization: the *variety of display formats* available to the commander, the ability to *navigate among these displays* in an unrestricted manner, and the ability to *query* the VIEW prototype and *highlight* display areas that satisfy particular parameters.

The VIEW prototype provides nine distinct display formats to capture the complexity of battlefield mental representations and mental models. The display formats include: maps and overlays, bar graphs, decision-trees, synchronization matrices, unit hierarchies, organization charts, and a variety of dialogue boxes and text windows. Different formats emphasize different aspects of the domain, the tasks, and the commander's inferencing. The basic display formats can be modified by the commander to reflect the specifics of a particular situation. Each display emphasizes a different combination of display/mental model parameters and thus different displays

are suited for different types of inferencing and information integration. Examples of the individual display formats are described below.

A key display format in VIEW is the familiar *map and overlay* display, which is currently the predominant graphical format used by the army commanders. The combined map+overlay displays have a number of advantages: they represent a large amount of information in a readily understandable, familiar format; they combine spatial representations (which trigger lower-level perceptual processing) with abstract symbology (which trigger higher-level symbolic processing), thus providing both an overall context (e.g., map of an entire area) and a specific aspect of the situation on which to focus (e.g., arrows representing movement, icons representing units and weapons; etc.).

The *bar graph* represents an efficient and effective means of rapidly displaying the same type of information (e.g., remaining or required quantity) about a number of different variables (e.g., different resources). The format of the display lends itself to a fast assimilation of the relative status of a large number of variables and anomalies can be identified quickly and in a single scan.

While new display formats can capture a unique way of viewing information, in many cases an enhancement of an existing display format is sufficient to create a powerful means of filtering and combining relevant information. A *hierarchical depiction of the unit composition* is an example of such a display format. The familiar hierarchy provides an overall context, allowing the commander to view units at different levels of hierarchy in the same "scan", and providing a display background on which a variety of information (i.e., different characteristics of the particular unit) can be overlayed (e.g., weapons and resources available, level of combat readiness, etc.).

Another hierarchical display, the *decision tree*, is unique in that it combines a trace of a cognitive process over time; namely, it provides a trace of the decision making process with respect to the development of a particular COA sequence. Time is thus an implicit dimension in this display. Furthermore, the display is highly abstract and symbolic, depicting a series of complex situations by a single labeled node in a tree diagram. As such, this display is well suited as a type of navigation backbone, through which to access the variety of other displays and information available about the situation.

The *navigation component* of the prototype facilitates unrestricted movement between the different display formats by allowing the commander to view displays containing identical objects or displays depicting related relevant information.

A critical component of VIEW prototype is the support it provides for *automatic detection of specific conditions* of the terrain, units, resources, or overall situation that might be of interest during planning. These conditions are expressed either as queries to the system or as rules defining some alarm or alert condition or a general situation of interest. Queries and rules are used to represent situations that might be desirable or undesirable and are a means of automatically detecting particular situations and displaying relevant information to the commander. Queries and rules thus serve the function of an *intelligent assistant,* who is aware of particular conditions which the commander should be aware of and notifies the commander when conditions occur. In the VIEW prototype the queries and rules thus allow the commander to explicitly visually represent important tactical decision making information combined into a single high-level construct. Examples of such constructs were elicited from the SMEs using repertory grid analysis.

The design of the VIEW prototype provides the knowledge engineer with a wide variety of tools to support the process of knowledge elicitation, the subsequent data analysis, and the

final interpretation of the results, where necessary. The VIEW design provides an environment within which a variety of knowledge elicitation techniques can be performed, both direct and indirect, and a variety of data collection methods can be employed to support these techniques. The knowledge elicitation component of the design is tightly coupled with the visualization component, and thus the full-functionality of the visualization component is available to the knowledge engineer and the subject matter expert. The user (knowledge engineer or subject matter expert) interacts with the VIEW prototype via graphical user interface, which contains a number of screens that support a variety of knowledge elicitation techniques. The existing design demonstrates a sequence of mock-up interface screens and indicates how these would be used during an elicitation session.

Specifically, the VIEW elicitation design provides the user with a variety of graphical user interfaces. The prototype design includes the following functionalities:

Graphical Displays and Visualizations

- A library of graphical displays at varying levels of complexity which can support both direct and indirect elicitation.
- Support for a variety of data collection techniques through the systematic presentation of displays and stimuli to the SME to elicit both qualitative and quantitative judgments.

Direct Elicitation Techniques

- Facilities for entering and analyzing free-form text while viewing different displays for a particular scenario.
- Facilities for constructing and editing domain vocabularies and concept maps during the elicitation session.
- Facilities for constructing aggregate structures from these domain primitives to reflect the experts' mental models.
- Facilities for editing and browsing the elicited structures.

Indirect Elicitation Techniques

- Facilities for editing and transformation of the elicited data.
- A repertoire of statistical techniques for analysis.
- A flexible environment for displaying the analyzed data and for assisting with the interpretation process.

The direct knowledge elicitation techniques, case-based display-centered interviews and decision-centered interviews, all provided the data for defining the critical elements of the visualization architecture: object definitions (e.g., terrain, terrain types, environmental objects, map overlays, military templates for depicting situations and decision-making, etc.), display definitions (e.g., maps and overlays, synchronization matrices, decision trees, bar graphs, process diagrams, etc.), and query and rule definitions (e.g., definitions of specific constraints representing high-level cognitive and perceptual constructs of interest to the commander). In addition to these data, the display-centered techniques provided information about the desired types of displays and their use during battlefield visualization. Examples of desired display types and functionalities included the following: ability to view a 3-D terrain representation from arbitrary perspectives, ability to combine and display a variety of weapons and electronic equipment characteristics, ability to support wargaming and what-if simulations through animation, automatic overlay and comparison of event and situation templates to quickly detect differences between predicted and actual situations, and the ability to zoom within an area and rapidly move among different levels

of abstraction. Due to the limited scope of this initial effort many of the suggested display formats and display manipulations could not be implemented. However, the information generated using the display-centered elicitation method is included in the recommendations for the follow-on Phase II effort.

The indirect knowledge elicitation effort, which focused on repertory grid analysis, yielded a number of classification attributes relevant for battlefield visualization. These attributes were elicited using different courses of action and different corridors of mobility. Examples of elicited classification attributes are: fire power deployable, concealment sensitivity to season, possibility of destroying concealment, maneuverability in bad weather, maneuverability in reduced visibility, safety in reduced visibility, vulnerability to ambushes, areas of vulnerability within corridor, ability to conceal rate of movement, ability to conceal number of troops, and ability to conceal exact location.

While some of the attributes were also obtained through direct elicitation, the repertory grid method generated a large number of complex constructs quickly and easily. We therefore recommend it as an effective and efficient means of obtaining complex cognitive and perceptual constructs. Our experience with using just two entity types for comparison and generating over 60 attributes, many of which represent complex tactical constructs, indicates that repertory grid analysis is a powerful technique for eliciting the commander's mental model attributes and warrants further exploration. A major feature of the elicitation component of the VIEW prototype design is a flexible means of presenting graphical entities for comparison during the initial stages of the repertory grid process. The VIEW prototype thus promises to be a powerful tool for eliciting a wide variety of tactical constructs, which can then be translated into visual format using the visualization component of the VIEW prototype.

Following our prototype demonstration, we specified the requirements for full-scope development of the VIEW concept, under a Phase II design, development, and validation effort. Under Phase I, the objective was to establish feasibility; under Phase II we would considerably expand the scope, increase the functionality of the modules, and fully explore the tool's utility in a formal validation exercise. The system architecture would follow that established by this Phase I study, but the functionality of the individual component modules would be considerably expanded. In particular, the *object world model* would be expanded to provide for dynamic simulation of friendly/enemy mobility, and limited computer-based wargaming. The object database would undergo considerable expansion in both the types of objects represented, and in the fidelity of representation. This would include all three object classes now represented in the Phase I model: terrain objects, military unit objects, and ground environment (operational) objects. The *visualization module* would also be expanded, to account for a greater range of conventional military displays, as well as an expandable set of unconventional displays subserving effective mental model representation. The *knowledge elicitation* module would be extended considerably beyond the user interface design, and include full functionality both in the interface, and in the underlying analysis software libraries. A direct linkage to the object database would also ensure that a "snapshot" of the actual tactical situation was available, to support the development of context-dependent user activity models.

We believe that these results demonstrate the basic features of the VIEW concept for mental mode visualization, elicitation, and refinement, particularly as applied to the commander's mental model of the battlefield. The study was specifically structured to be narrow in scope, but of sufficient depth to ensure the reliable specifications of requirements for a full-scope system.

5.3 Recommendations

On the basis of these Phase I results, a Phase II effort is recommended which focuses on the design, development, and validation of a full-scope prototype Visualization and Interactive Elicitation Workstation (VIEW) for inferring the commander's mental model of the battlefield. Table 5 highlights the basic differences between the completed Phase I effort and the recommended Phase II program, and shows how results of the Phase I effort feed into the objectives of the Phase II program.

The overall objective of the Phase I effort was to assess feasibility of the VIEW concept, by evaluating enabling technologies, developing the concept prototype, and demonstrating its operation on a desktop computer. The Phase II objective is to expand upon the Phase I design and develop a full-scope VIEW prototype with enhanced capabilities over a broader range of scenarios. The scope of the Phase I effort was limited to a single scenario incorporating limited branching capability. Under Phase II, we would expand the scope to support multiple scenario types and intensity levels while providing for a more general branching capability to evaluate alternative courses of action, and unexpected evolution in the tactical situation. The approach used in Phase I relied on design of the VIEW concept, and development and informal demonstration of a prototype for a well-defined scenario. Under Phase II, we would enhance the VIEW prototype, conduct extensive demonstrations, and follow up with a formal field evaluation in a well-defined knowledge elicitation exercise.

The system architecture developed in Phase I was designed to support concurrent visualization and elicitation functions, integrated via a common object database. In Phase II we intend to support the same basic functionality, but with extensions in scope across all modules, especially the object world model.

The system components include: 1) object world model; 2) object database; 3) visualization module; 4) and knowledge elicitation module. Under Phase II we plan to maintain the same module structure, but with significant enhancements to each, to support full-scope operation.

In the Phase I effort, the object world model was implemented via a simple preprogrammed script, constructed for demonstration purposes. No provision was made for dynamic simulation of friendly/enemy mobility, nor was any provision made for computer-based wargaming, although the latter could be accomplished via manual modification of friendly/enemy disposition; however, this was not the focus of the Phase I effort. Under Phase II we plan to expand the capabilities of the object world model by providing for generic military unit *behaviors*, which will support dynamic simulation as well as limited wargaming capabilities.

Under Phase I the object database provided the common object representation for all visualization/elicitation components of the system. Under Phase II we plan to keep this architecture but considerably expand the individual types and number of objects represented in the database. In the Phase I prototype the object database consisted of three general classes: Terrain Objects, Military Unit Objects, and Ground Environment Objects. Under Phase II we intend to expand these classes to include other key military concepts, terms and symbols.

Under Phase I the Terrain Object database included topographic features, bridges/hydrographics, communications, boundaries and the prototype grid system. Under Phase II we plan to expand on all these sub-classes to provide the system with more terrain detail. Under Phase I, topographic objects included elevation, slope contours, and vegetation. Under Phase II we will implement object databases in this sub-class to include historical preservation areas and other topographic features. Under Phase II, we plan to add canals and waterfalls to the

hydrographic object database. Under Phase I, the Terrain Object databases represented key communications features and boundaries such as roads and railroads, as well as political boundaries. In Phase I, the grid system was selected to simply support the demonstration; under Phase II we would convert this to a military grid system.

In Phase I the object database represented three levels of Military Units: brigade, battalion, and company. In Phase II we would extend this downward to the platoon level, and in special cases provide for squad representation. Under Phase I unit types can be selected from a list of seven options: infantry, armor, air defense artillery, field artillery, aviation, engineering, and special forces. Under Phase II, we propose to keep the same structure, but add to the sub-classes already available in the VIEW prototype. For example, the engineering unit type does not have any subclasses; under Phase II we plan to include sub-classes such as amphibious engineering as an option for the user. In Phase I, the composition and organization of units included organic units with its composition, and certain types of attached units. The VIEW prototype under Phase I allows for task organization for one level. Under Phase II, we plan to allow for task organization at all levels.

In Phase I, the object database section covering Ground Environment Objects included Areas of Operation, Area of Interest, Avenues of Advance and Approach, and Unit boundary delineation. Under Phase II, the expansion of the Ground Environment Object database will include symbols such as military points, lines, areas, routes, obstacles, crossings, and tactical deception objects. In addition, under Phase II these objects will be extended to include other ground environment entities such as movements, fire planning, and battlefield activities. Under Phase II, we would further extend this feature set to include other military ground environment entities such as installation role indicators, equipment indicators, and communication and electronic emitters.

In the Phase I effort, the visualization module provided support for nine basic types of displays: dialog box, 2-D topographic, unit graphical, organizational chart, bar graph, synchronization matrix, decision tree, textual, and process diagram. Under Phase II we intend to support the same basic types, but with extended coverage across the terrain and units, as well as with enhanced interfaces to better visualize and modify the component objects. In addition we propose to add animation where it will support dynamic visualization, and also provide for the presentation of remote sensing data. In Phase I we provided a fixed library of display types. Under Phase II we would provide a display editing functionality which would enable the user to construct new display types to reflect the emerging structure of the mental representations. In Phase I the navigation mode across visualizations was menu-selectable and context-sensitive. Under Phase II we propose to use the same technique but with stronger contextual filtering to support more rapid navigation by the user. Under Phase I we provided for two query modes, a general Boolean query useful for accessing attributes regarding individual objects, and a menu-based query used for obtaining information regarding terrain areas. Under Phase II we propose to develop an integrated Boolean query that could be applied to either objects or general terrain regions, or both. Under Phase I dynamic updating of the visualization interfaces was limited to the 2D topographic displays, the unit graphical displays, and the decision tree. Under Phase II we propose to extend dynamic updating to all graphical objects, to reflect the changes occurring in the underlying database.

In the Phase I effort the functionality of the knowledge elicitation (KE) module was limited to the design of the user interface, a mock-up of window sequences during knowledge elicitation, and specifications for functionalities supporting both direct and indirect elicitation. Under Phase II we propose to extend this interface to support additional direct and indirect KE

techniques. We also propose to provide an open architecture for direct addition of other techniques that may be of interest to KE researchers. In Phase I the recording of the KE data was accomplished manually. In Phase II we propose to maintain this manual interface, but extend it to also include the automatic recording of the user's system use, including both display and control, as well as the status of the database, to insure an ongoing "snapshot" of the actual tactical situation as the knowledge is elicited from the user. This will support the development of a context-dependent user activity model. Under Phase I data types for direct KE were limited to a variety of objects (including both concrete objects such as terrain elements and military equipment, and abstract objects, such as decision points), and their attributes (which included complex tactical constructs), which were combined into hierarchical structures representing unit hierarchies and decision trees depicting the planning process. For indirect KE they included rankings and similarity/difference assessments. Under Phase II we plan to extend the object types to include abstract tactical objects reflecting the complex constructs elicited via indirect knowledge elicitation techniques. We also plan to extend the types of aggregate object structures to include causal models, procedure trees, and generalized concept maps as an intermediate representation from which to construct specialized structures. In addition to the basic data types available in Phase I, we plan to include basic psychometric measurements such as reaction times, error ratings, recall measures, and metrics of situation assessment and decisionmaking performance. In Phase I the analysis capability was supported by a post-session download to external statistical programs, and no provision was made in the Phase I prototype for on-line analysis. In Phase II we propose to provide for this capability to support on-line analysis for interactive guidance of the scenario and visualization choices, as the KE analysis results become available.

Under Phase I the validation and demonstration effort was limited to a single-string scenario, with the primary focus emphasizing evaluation of overall feasibility of the VIEW concept, and general reasonableness of the results. Under Phase II we will evaluate VIEW operation with multiple scenarios and dynamic branching providing for extensive scenario modification and evolution. Under Phase I, evaluation was primarily via two subject matter experts, conducting informal evaluations of VIEW prototype functionality and utility. Under Phase II we intend to formalize this process using an SME panel, guided by formal assessment metrics of VIEW utility. In Phase I our demonstration was limited to a demonstration of basic capabilities and an identification of potential growth paths in functionality. Under Phase II we intend to demonstrate VIEW operation in multiple well-studied scenarios and formally identify its capabilities and limitations.

The Phase I design and implementation of the VIEW prototype specified the overall architecture incorporating a linkage to the visual objects. Under Phase II we propose a full implementation of the VIEW prototype with extensions to the objects to provide for full system functionality. Under Phase I we provided for multiple concurrent windows and context-dependent navigation across those visualizations. Under Phase II we intend to follow the same basic design with extensions to support further visualizations and greater ease of navigability. The software implementation in Phase I used Microsoft Windows, Microsoft Visual Basic, Microsoft Access, and Visio Technical. Under Phase II we propose development in a more full featured language including Visual C++, which would support complex computations, efficient operation, and a sophisticated graphic interface. In addition we would propose the use of CLIPS to support rapid development of a forward-chaining expert system, for rapid rule

Table 5. Features of phase I and phase II efforts

Feature	Phase I	Phase II
• Objective	• Establish Feasibility of Hybrid Methodology	• Develop Full-Scope Prototype Visualization/Elicitation System
• Scope	• Single Scenario with Limited Branching Capability	• Multiple Scenario Types & Intensity Levels • General Branching Capability
• Approach	• Concept Prototype Development & Demonstration	• Prototype Enhancement, Demonstration, & Field Evaluation
• System Architecture	• Visualization & Elicitation Modules Integrated via Object Database	• Same as Phase I, with Object World Model Extension
• System Components	• Object World Model, Object Database, Visualization Module, KE Module	• Same as Phase I
• Object World Model	• Preprogrammed script for demonstration • Provision for manual wargaming	• Dynamic simulation of friendly/enemy units • Limited wargaming simulation
• Object Database:		
- Terrain		
- Topographic features	- Elevation, slope contours, vegetation	- Add historical preservation areas, include wildlife,...
- Bridges / Hydrographics	- Rivers, bridges	- Add canals, waterfalls,...
- Communications	- Roads, railroads	- Add tunnels, footpaths,...
- Boundaries	- None	- Add political, religious,...
- Grid System	- Demonstration system	- Add military grid system
• Object Database: -Military Unit	- Three Levels: brigade, battalion, company - Organization: constituent and supporting units - Types: mechanized infantry, mechanized armor, etc. - Qualitative factors: training level, recent replacement, recent activity	- Extend to platoon - Extend to special forces - Extend to combined arms - Extended to other combat readiness metrics

Table 5. Features of phase I and phase II efforts (continued)

- Weapon systems	- limited to TOW, Tanks, DRAGONS, Mortar, etc.	- Extended to a broader range of weapon systems; include weapon characteristics such as range,...
• Object Database: - Ground Environment	- Battlefield area includes AO/AI - Avenues of advance/approach - Unit boundary delineation	- Include all Line, Area, Route, Obstacles, Crossings, Movements, Fire Planning, ..
• Visualization Module - Types	- 2D Topographic - Icons/Symbols - Organizational Chart - Bar Graph - Synchronization Matrix - Decision Tree - Textual - Process Diagram	- Same as Phase I, but with extended coverage & enhanced interfaces - Add animation, remote sensing data
- Display Design	- Limited to fixed display types	- Extensive display type library through a display editing functionality
- Navigation Mode	- Menu selectable	- Menu selectable; context dependent
- Query Mode	- Boolean query & menu selection	- Integrated Boolean query
- Dynamic Updating	- Limited to 2D Topographic, Unit Icons, and Decision Tree	- Extend to all graphical objects in Visualization Module
• Knowledge Elicitation Module - Interface	- Text recording for DKE techniques - Text/numeric recording for IKE techniques, menu-driven	- Extend to additional DKE and IKE techniques, Open architecture for technique addition
- Data recording	- Manual via KE windows & menus	- Manual interface - Record user's display use (proximity data) - Link to object database
- Data types	- Proximity assessments, rankings & similarity/difference assessments	- Include RT's error rates, recall measures, & SA/DM measures
- Analysis capability	- Post-session download to external programs	- On-line analysis & display, with interactive guidance of scenario and/or visualization

Table 5. Features of phase I and phase II efforts (continued)

• Validation/ Demonstration - Validation & Test - Demonstration	- Limited single-string scenario - Informal evaluation by two SME's - Demonstrate basic capabilities & growth path	- Multiple scenarios with dynamic branching - Formal evaluation by SME panel - Demonstrate in multiple well-studied scenarios
• Design & Implementation - Prototype - Interface - Implementation	- Specify architecture with database linkage to visual objects - Multiple windows, with context-dependent navigation - Microsoft Windows, Visual Basic, Access, & Visio Technical	- Full implementation of prototype, with extensions to prototype objects - Extension of Phase I design - Same as Phase I, with extended object library in Visual C++, CLIPS for rule evaluation

References

Adams, M. J., Tenney, Y. J., & Pew, R. W. (1995). Situation awareness and the cognitive management of complex systems. Human Factors, 37(1), 85-104.

Badre, A. N. (1978). Selecting and representing information structures for battlefield decision systems (ARI Technical Report 79-A20). Alexandria, VA: U.S. Army Research Institute for the Behavioral and Social Sciences.

Berry, D. C. (1987). The problem of implicit knowledge. Expert Systems, 4(3).

Boose, J. H. (1985). A knowledge acquisition program for expert systems based o personal construct psychology. International Journal of Man-Machine Studies, 23, 495-525.

Bradshaw, J. M., Ford, K. M., Adams-Webster, J. R., & Boose, J. H. (1993). Beyond the repertory grid: New approaches to constructivist knowledge acquisition tool development. International Journal of Intelligent Systems, 8, 287-333.

Brezovic, C. P., Klein, G. A., Y Thorsden, M. (1987). Decision making in armored platoon command, interim report (ARI Research Note 90-51) (KATR-858-(B)-87-05F). Alexandria, VA: U.S. Army Research Institute for the Behavioral and Social Sciences. (AD A231 775)

Broadbent, D. E., Fitzgerald, P., & Broadbent, M. H. P. (1986). Implicit and explicit knowledge in the control of complex systems. British Journal of Psychology, 77, 33-50.

Castro, F. D., Jr., Hicks, H. E. Jr., Ervin, J. R., & Halpin, S. M. (1992). ACCES application of 91-02: ACCES assessment of command and control during a division level CPX (ARI Research Note 92-78). Alexandria, VA: U.S. Army Research Institute for the Behavioral and Social Sciences. (AD A256 433)

Chi, M. T. H., Glaser, R., & Farr, M. J. (1988). The nature of expertise. Hillsdale, NJ: Erlbaum.

Cooke, N. M., Durso, F. T., & Schvaneveldt, R. W. (1986). Recall and measures of memory organization. Journal of Experimental Psychology: Learning, Memory, and Cognition. 12, 538-549.

Cooke, N. M., & McDonald, L. E. (1988). The application of psychological scaling techniques to knowledge elicitation for knowledge-based systems. In J. Boos and B. Gaines (Eds.), Knowledge acquisition tools for expert systems. New York: Academic Press.

Deckert, J. C., Entin, E. B., Entin, E. E., MacMillan, J., & Serfaty, D. (1994). Military command decisionmaking expertise (Report 631). Alphatech.

deKleer, J., & Brown, J. S. (1983). Assumptions and ambiguities in mechanistic mental models. In Gentner and Stevens (Eds.), Mental models. Hillsdale, NJ: Erlbaum.

Detterman, D. K. (1980). Models of mental functioning. Final report (Technical Report 890813). Bolling AFB, DC Air Force Office of Scientific Research.

Endsley, M. R. (1995). Toward a theory of situation awareness in dynamic systems. Human Factors, 35(1), 32-45.

Ericsson, K. A., & Simon, H. A. (1984). Protocol analysis: Verbal reports as data. Cambridge, MA: The MIT Press.

Fallesen, J. J. (1993). Overview of Army tactical planning performance research (ARI Technical Report 984). Alexandria, VA: U.S. Army Research Institute for the Behavioral and Social Sciences. (AD A273 273)

Fallesen, J. J., Carter, C. F., Jr., Perkins, M. S., Michel, R. R., Flannagan, J. P., & McKeown, P. E. (1992). The effects of procedural structure and computer support upon selecting a tactical course of action (ARI Technical Report 960). Alexandria, VA: U.S. Army Research Institute for the Behavioral and Social Sciences. (AD A257 254)

Fallesen, J. J., & Michel, R. R. (1991). Observation on command and staff performance during CGSC Warrior '91 (ARI Working Paper LVN-91-04). Alexandria, VA: U.S. Army Research Institute for the Behavioral and Social Sciences.

Forbus, K. (1983). Qualitative reasoning about space and motion. In Gentner and Stevens (Eds.), Mental models. Hillsdale, NJ: Erlbaum.

Gaines, B., & Boose, J. (1988). Knowledge acquisition for knowledge-based systems. New York: Academic Press.

Gentner, D., & Gentner, D. R. (1983). Flowing waters or teeming crowds: Mental models of electricity. In Gentner and Stevens (Eds.), Mental models. Hillsdale, NJ: Erlbaum.

Gentner, D., & Stevens, A. L. (1983). Mental models. Hillsdale, NJ: Erlbaum.

Hammond, K. R., Hamm, R. M., Grassia, J., & Pearson, T. (1987). Direct comparison of the efficacy of intuitive and analytical cognition in expert judgment. IEEE Trans. on Systems, Man and Cybernetics, 17(5), 753-770.

Hegerty, M., Just, M. A., & Morrison, I. R. (1988). Mental models of mechanical systems (54). Carnegie-Mellon University.

Hudlicka, E., & Huggins, A. W. F. (1994). The use of cognitive science techniques for developing FAA inspection indicators final technical report (BBN Report 7940). Cambridge, MA: BBN Laboratories.

Johnson, S. C. (1967). Hierarchical cluster analysis. Psychometrika, 32(3, September).

Johnson-Laird, P. N., Byrne, R. M. J., & Schaeken, W. (1992). Prepositional reasoning by model. Psychological Review, 99(3), 418-439.

Kelly, G. (1955). The psychology of personal constructs. New York: Norton Publishers.

Kieras, D. E. (1984). A simulation model for procedure inference from a mental model for a simple device (15). Tucson, AZ: University of Arizona.

Klein, G. A. (1989a). Recognition-primed decision. Advances in Man-Machine Systems Research, 5, 47-92.

Klein, G. A., Calderwood, R., & Clinton-Cirocco, A. (1986). Rapid decision-making on the fire ground. Proceedings of the Human Factors Society 30th Annual Meeting.

Klein, G. A., Calderwood, R., & MacGregor, D. (1989). Critical decision method for eliciting knowledge. IEEE T-SSMC, 19(3), 462-472.

Knaueper, A., & Rouse, W. B. (1984). A rule-based model of human problem solving behavior in dynamic environments. IEEE T-SMC, 15, 708-719.

Kosslyn, S. M., Cave, C. R., Forbes, Z. E., & Brunn, J. L. (1983). Imagery ability and task performance (2). Brandeis University.

Kosslyn, S. M., & Schwartz, S. P. (1977). A simulation of visual imagery. Cognitive Science, 1(3).

Larkin, J. H. (1983). The role of problem representation in physics. In Gentner and Stevens (Eds.), Mental models. Hillsdale, NJ: Erlbaum.

Larkin, J. H., & Simon, H. A. (1987). Why a diagram is (sometimes) worth ten thousand words. Cognitive Science, 11, 65-99.

Leddo, J. M., & Cohen, M. S. (1989). Cognitive structure analysis: A technique for eliciting the content and structure of expert knowledge. Proceedings of the 1989 AI Systems in Government Conference.

Lewis, H. V., & Fallesen, J. L. (1988). Human factors guidelines for command and control systems: Battlefield and decision graphics guidelines (ARI Report 89-01). Alexandria, VA: U.S. Army Research Institute for the Behavioral and Social Sciences.

Lohse, G. L., Biolsi, K., Walker, H., & Rueter, H. H. (1994). A classification of visual representations. Communications of the ACM, 37(12), 36-49.

Lussier, J. W., Solick, R. E., & Keene, S. D. (1992). Experimental assessment of problem solving at the Combined Arms Services Staff School (ARI Research Note 92-52). Alexandria, VA: U.S. Army Research Institute for the Behavioral and Social Sciences. (AD A253 386)

McDonald, J. E., & Schvaneveldt, R. W. (1987). The application of user knowledge to interface design (Technical Report MCCS-87-93). Las Cruces, NM: New Mexico State University, Computing Research Laboratory.

Nisbett, R. E., & Wilson, T. D. (1977). Telling more than we know: Verbal reports o mental processes. Psychological Review, 84, 231-259.

Norman, D. (1983). Some observations on mental models. In Gentner and Stevens (Eds.), Mental models. Hillsdale, NJ: Erlbaum.

Olson, J. R., & Biolsi, K. (1990). Techniques for representing expert knowledge (Technical Report 34). University of Michigan, Cognitive Science and Machine Intelligence Laboratory.

Roschelle, J., & Greeno, J. G. (1987). Mental models in expert physics reasoning (GK-2). University of California at Berkeley.

Rouse, W. B., & Miller, N. M. (1986). On looking into the black box: Prospects and limits in the search for mental models. Psychological Bulletin, 100(3), 349-363.

Salter, W. (1983). Tacit theories of economics. 5th Annual Conference of Cognitive Science Society, Rochester, NY.

Sarkar, M., & Brown, M. H. (1994). Graphical fisheye views. Communications of the ACM, 37(12), 73-84.

Schank, R. C. (1983). Dynamic memory: A theory of reminding and learning in computers and people. New York: Cambridge University Press.

Schmidt, H. G., Boshuizen, H. P. A., & Hobus, P. P. M. (1988). Transitory stages in the development of medical expertise: The "intermediate effect" in clinical case representation studies. 10th Annual Conference of the Cognitive Science Society, Montreal, Canada.

Shaw, J. J., & Powerll, W. S. (1989). Lessons learned from the Cascade polar exercise (Technical Report). Alphatech.

Tolcott, M. A., Marvin, F. F., Y Lehner, P. E. (1989). Expert decisionmaking in evolving situations. IEEE Trans. on Systems, Man and Cybernetics, 19(3), 606-615.

Tufte, E. R. (1983). The visual display of quantitative information. Chesire, CT: Graphics Press.

Walter, J. C., & Warren, P. T. (1992). JANUS-3D providing three-dimensional displays for a traditional combat model. Monterey, CA: Naval Postgraduate School Master's Thesis.

White, B. Y., & Frederiksen, J. R. (1987). Causal model progressions as a foundation for intelligent learning environments (BBN Report 6686). Cambridge, MA: BBN Laboratories.

Witte, T., & Kelly, V. C. (1994). Visualization support for an Army reconnaissance mission (95-43). Fort Belvoir, VA: Army Topographic Engineering Center. (DTIC Technical Report 213)